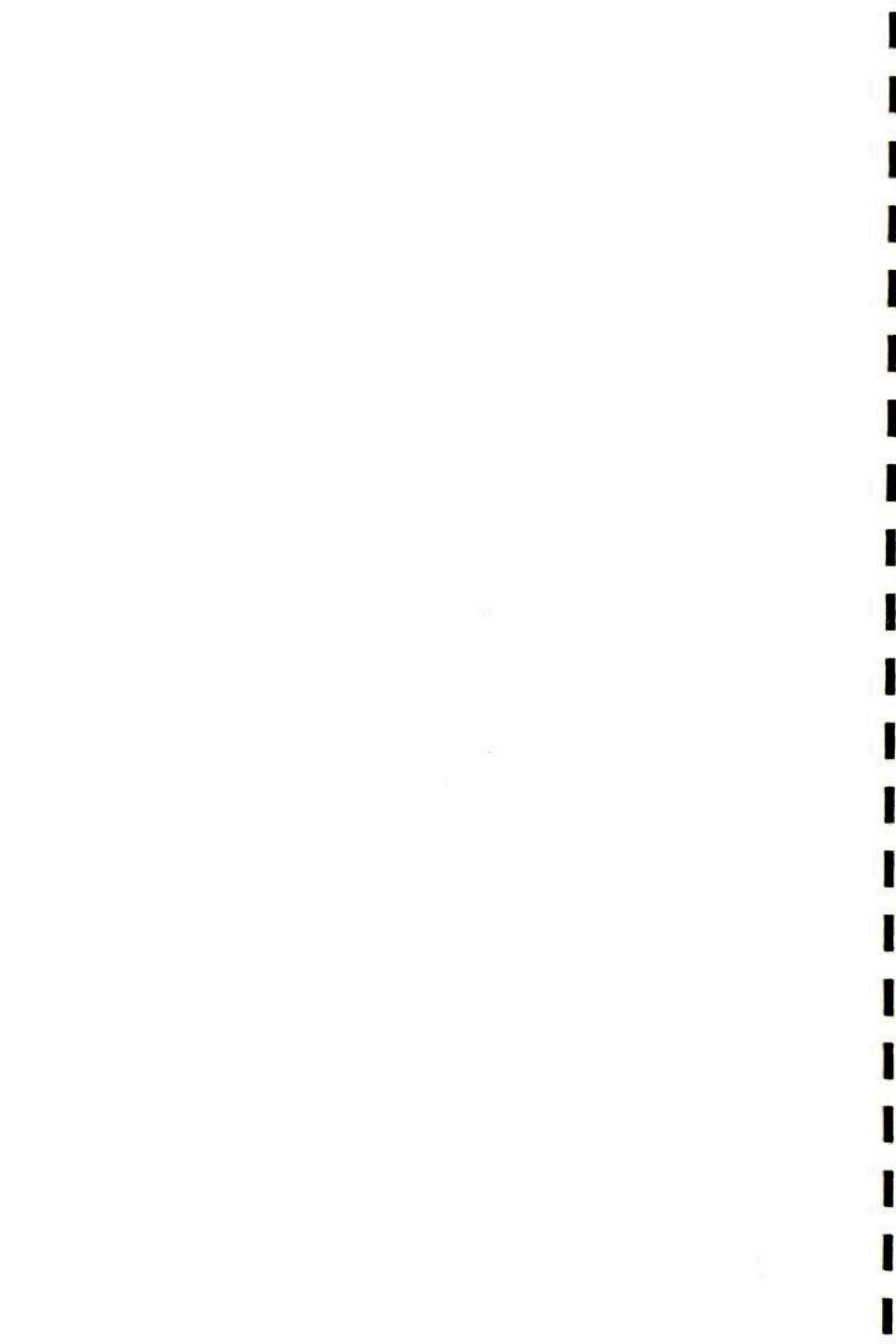
BASIC-PROGRAMMA'S VOOR HET HELE GEZIN

Max Voorburg

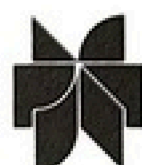
Kluwer Software-reeks



BASIC-programma's voor het hele gezin

Max Voorburg

BASIC-programma's voor het hele gezin



Kluwer Technische Boeken B.V. Deventer — Antwerpen

Omslag en illustraties: Wim Niessink

ISBN 90 201 1702 5

D/1983/0108/217

1e druk 1983

2e oplage 1984

3e oplage 1984

Woord vooraf

Dit boek is gemaakt met als doel de gebruiker van microcomputers een aantal leuke, maar ook zinvolle programma's in handen te geven. De programma's zijn geschreven in standaard BASIC. Dat wil zeggen dat er op gelet is dat deze programma's op 95% van alle computers met een BASIC-interpreter of -compiler kunnen draaien. Toch bestaat de mogelijkheid dat er hier en daar kleine aanpassingen noodzakelijk zijn. Achterin het boek is een uitgebreide appendix opgenomen met zo goed als alle BASIC-statements en -functies met hun equivalenten voor andere machines. Een aantal kleine probleempjes zal hier besproken worden met daarbij een oplossing of een aanwijzing.

In de educatieve programma's en de spelprogramma's wordt vaak gebruik gemaakt van willekeurige getallen. Deze willekeurige getallen worden gegenereerd door een zogenaamde randomgenerator, een stukje programma. Deze randomgenerator moet gestart worden om ook werkelijk willekeurige getallen te produceren. Dit starten geschiedt met het statement RANDOM, zoals dat bijvoorbeeld voorkomt op de TRS-80. Andere computers kennen statements als RAND of RANDOMIZE. Beschikt men niet over zo'n statement dan kan men het ook weglaten. De kans bestaat dan echter dat de randomgenerator steeds bij het zelfde getal start. Wanneer men dat wil voorkomen, dan kan men het RANDOM-statement vervangen door een klein programma.

```
10 PRINT "GEEF GETAL < 1-20> ";
20 INPUT A
30 FOR I = 1 TO A
40 B = RND(0)
50 NEXT I
```

Nu dient zich onmiddellijk het tweede probleem aan. In de verschillende BASIC-versies is de RND-functie niet gelijk. Om dit probleem nu zoveel mogelijk te ondervangen is in dit boek van een standaard uitgegaan. In alle programma's waarin de RND-functie wordt gebruikt, staat RND(0).
De functie RND(0) levert een getal op dat ligt tussen nul en een. Zo kan bijvoorbeeld de functie RND(0) het getal 0.76212 of 0.23498 opleveren. In een aantal BASIC-interpreters dient de functie niet aangeroepen te worden met RND(0) maar met RND(1) om het zelfde resultaat te krijgen, dit geldt bijvoorbeeld voor de Philips P-2000.
Om het scherm in een keer te wissen en de cursor links boven in de hoek (homepositie) te krijgen wordt in deze programmatuur gebruik gemaakt van het statement CLS. Dit betekent CLEAR SCREEN. Bij een aantal computers komt dit statement voor, maar bij andere weer niet. De gebruikers van bijvoorbeeld een Exidy Sorcerer zullen PRINT CHR$(12) moeten ingeven in plaats van CLS. Bij de Commodore computers is dit CLEAR/HOME.
Bij de ontwikkeling van de programmatuur is ervan uitgegaan dat het beeldscherm wordt gebruikt als uitvoereenheid en het toetsenbord als invoereenheid. De verleiding was groot om bij een aantal programma's de cassette of floppy disk als achtergrondgeheugen te gebruiken. Uiteindelijk is hiervan afgezien omdat de communicatie tussen computer en achtergrondgeheugen niet uniform geregeld is.

Een van de grote problemen bij het beeldscherm is de lengte van de regels. Bij de samenstelling van dit boek is ervan uitgegaan dat er maximaal 40 karakters op een regel kunnen en minimaal 14 regels onder elkaar voordat het scherm omhoog wipt.
Sommige programma's worden uitvoerig ingeleid, zodat de gebruiker in staat is snel de werking van een programma te begrijpen of hoe een interne data-file in elkaar zit. Later kan hij deze interne data-file met eigen gegevens vullen. Tegelijkertijd worden bij de programma's suggesties gegeven om de programmatuur te veranderen of aan te passen aan de eigen machine. Gezien de strakke gestructureerde modulaire opbouw hoeft dit geen probleem te zijn.

Inhoud

1. Inleiding

De programma's in dit boek zijn opgezet met de gedachte dat zowel beginners, als gevorderden deze moeten kunnen gebruiken. In het kort zal nu worden uiteengezet hoe de programmatuur is opgebouwd en wat voor standaardoplossingen er gebruikt zijn.
Het boek bestaat uit drie gedeelten. Het eerste gedeelte bestaat uit wat men zou kunnen noemen educatieve programma's. Dit zijn programma's voor kinderen van ongeveer vier tot twaalf jaar. Het tweede gedeelte bevat spelprogramma's. Ze beginnen eenvoudig, maar lopen in moeilijkheidsgraad op. Het derde en laatste gedeelte van het boek wordt gevormd door algemene programma's. Hierin vindt men onder andere een programma om een catalogus mee bij te houden en een programma om hypotheken met elkaar te vergelijken.
Wanneer men het boek doorbladert, zal het opvallen dat de programma's wat structuur betreft erg veel op elkaar lijken. Zowel de eenvoudige als de ingewikkelde programma's zijn opgebouwd op eenzelfde gestructureerde modulaire manier. Dit is heel bewust gedaan; het bevordert de overzichtelijkheid en daardoor kunnen fouten razend snel opgespoord worden.
Op elke regel staat in de meeste gevallen maar één statement. Dit kost weliswaar meer geheugenruimte maar de programma's kunnen daardoor sneller worden ingetikt met minder kans op fouten. Op sommige regels staat achter het statement een dubbele punt. Deze dubbele punt wordt gevolgd door een REM-statement met een opmerking. Deze opmerkingen zijn toegevoegd ter verduidelijking. Men hoeft ze niet over te tikken. In principe kan men alle REM-statements weg laten. Dit heeft dan wel als nadeel dat het opsporen van tikfouten wordt bemoeilijkt. Ook later, wanneer men wijzigingen wil aanbrengen in het programma, zal men de REM-statements missen.
Alle programma's beginnen met een of meer NON-EXECUTABLE statements, dit zijn statements als RANDOM, DIM, DATA, etc. Deze statements zijn nodig voor de werking van het programma, maar de gebruiker merkt niet dat ze daadwerkelijk worden uitgevoerd. De NON-EXECUTABLE statements worden altijd gevolgd door het enige GOTO-statement in de programmatuur. Deze GOTO wijst naar een hoog regelnummer waar de stuurmodule wordt gestart. Deze stuurmodule houdt het gehele programma onder controle. Van hieruit worden de verschillende onderdelen van het programma (subroutines) aangeroepen en uitgevoerd.
Een voordeel is dat men de verschillende programmamodules door elkaar kan ontwikkelen. Men is niet meer gebonden aan een vaste plaats voor de programma-onderdelen. Een invoerblok hoeft dan niet meer vóór een uitvoerblok te staan. De gebruikte methode heeft ook een nadeel! Het nadeel is dat de programma's gemiddeld langer zijn dan bij de conventionele manier van programma-ontwikkeling. Daar staat dan wel tegenover dat deze manier van werken veel sneller resultaat oplevert.
Bij een tweetal programma's zal uiteengezet worden hoe men van idee tot het programma gekomen is. Het is voor een klein programma 'zet de getallen op volgorde', en voor een groot programma 'Zeeslag' gedaan.
Bij de opzet van de programmatuur is uitgegaan van een aantal basis-structuren. Het zou te ver voeren om alle structuren uit te leggen, maar de twee belangrijkste volgen hier. De twee structuren zijn belangrijk omdat zij helpen bij het snel opsporen van fouten en de gelegenheid bieden gemakkelijk wijzigingen in de programma's aan te brengen.

Het zal de gebruiker wel duidelijk zijn dat in de programmatuur vrij veel gebruik wordt gemaakt van het subroutinemechanisme. Bekend verondersteld mag worden, dat een subroutine aangeroepen wordt met GOSUB 'regelnummer' en dat de subroutine verlaten wordt met het RETURN-statement.
Elke subroutine voert in principe maar één, hooguit twee functies uit. Bijvoorbeeld de invoermodule haalt waarden op van het toetsenbord. De bewerkingsmodule manipuleert met variabelen en de uitvoermodule verzorgt het weergeven van de resultaten op het beeldscherm.
Een subroutine is dan ook te beschouwen als een klein programma. Zo'n klein programma heeft een begin en een eind. Het begin van de subroutine wordt gevormd door het eerst uit te voeren BASIC-statement. Dit statement heeft relatief gezien het laagste regelnummer in de subroutine. Het einde van de subroutine wordt gevormd door de RETURN-opdracht. De RETURN heeft het hoogste regelnummer in de subroutine. Op deze manier heeft elke subroutine een ingang en een uitgang. Daarbij beslaan subroutines maar een betrekkelijk klein aantal regels. Het zal duidelijk zijn dat fouten snel gelokaliseerd kunnen worden omdat in een subroutine slechts een beperkt aantal variabelen gemanipuleerd worden, binnen een beperkt aantal programmaregels.
FOR...NEXT-lussen willen in BASIC nogal eens aanleiding geven tot spaghetti-programmatuur. In dit soort lussen worden dan onnodig GOTO-statements geplaatst om uit de lus te komen, hetgeen de overzichtelijkheid niet ten goede komt. In dit boek wordt ervan uitgegaan dat FOR...NEXT-lussen maar één ingang hebben en één uitgang. Eigenlijk hetzelfde als bij de subroutines. Het begin van de FOR...NEXT-lus is de FOR..., het einde is de NEXT. Bij de NEXT wordt de lus verlaten. Voordeel ook hier weer, overzichtelijkheid. Nadeel, meestal heeft men iets meer programmaregels nodig. Om het een en ander te verduidelijken volgt hier een voorbeeld.
Voorbeeld: Matrix A is gevuld met tien getallen. De waarde van variabele K wordt ingevoerd via het toetsenbord. De ingevoerde waarde K moet voorkomen in matrix A. Komt de waarde voor dan vervolgt het programma zijn loop. Komt de waarde niet voor dan wordt de vraag herhaald.
Oplossing A is de oplossing, zoals deze in alle programma's in dit boek wordt gehanteerd. Oplossing B is een oplossing van de recht toe recht aan methode.

Oplossing A

```
10 PRINT "GEEF WAARDE ";
20 INPUT K
30 H = 0
40 FOR I = 1 TO 10
50 IF K <> A(I) THEN 80
60 H = I
70 I = 10
80 NEXT I
90 IF H = 0 THEN 10
100
```

Oplossing B

```
10 PRINT "GEEF WAARDE ";
20 INPUT K
30 FOR I = 1 TO 10
40 IF K = A(I) THEN 70
50 NEXT I
60 GOTO 10
70  .
    .
    .
```

Het verschil tussen oplossing A en B is duidelijk. Oplossing B is drie regels korter. Nadeel is wel dat er een GOTO in het programma sluipt. Erger dan de GOTO is wel dat de FOR...NEXT-lus twee uitgangen gekregen heeft. Eén uitgang op regel 50, de normale, en één in regel 40. In oplossing A heeft de FOR...NEXT-lus nog steeds één uitgang en wel in regel 80. Dat is de enige plaats waar de lus verlaten wordt. De werking van het programma is vrij eenvoudig en de structuur geldt voor alle FOR...NEXT-lussen in dit boek. In regel 30 wordt een hulpvariabele H gedefinieerd. Wanneer de waarde van deze variabele gelijk aan nul blijft, betekent dit dat de waarde van K niet in matrix A voorkomt. Is de waarde van H ongelijk aan nul, dan houdt dit in dat de waarde van K wel in matrix A voorkomt. In dit geval wordt H gevuld met de waarde van de index I. Tegelijkertijd wordt er voor gezorgd dat de index (de lus-teller) zijn maximale waarde krijgt, zodat de lus verlaten wordt.
Voordeel van oplossing A: overzichtelijkheid en uniformiteit. Nadeel meer programmaregels. Voordeel van oplossing B minder programmaregels, maar als groot nadeel onoverzichtelijkheid. Men weet niet wanneer en waar de lus verlaten wordt.
In een aantal programma's wordt gebruik gemaakt van DATA-statements om een interne data-file op te bouwen. De interne data-file kan twee functies vervullen. De eerste functie is die van opslaan van stuurgegevens voor een programma. Het programma Zeeslag is hiervan een voorbeeld. De tweede functie is die van achtergrondgeheugen. In de gevallen waarin duidelijk sprake is van achtergrondgeheugen, is de interne data-file achterin het programma na het END-statement opgenomen. Dit is gedaan om het aanvullen met extra DATA-statements te vergemakkelijken zonder dat de regelnummering van het feitelijke programma wordt aangetast.
Wanneer met een interne data-file als achtergrondgeheugen wordt gewerkt, is altijd een leesmodule aanwezig. Dit is gedaan om de mogelijkheid te scheppen de leesmodule te wijzigen zodat van een achtergrondgeheugen gelezen kan worden zonder daarbij het hele programma overhoop te hoeven gooien.

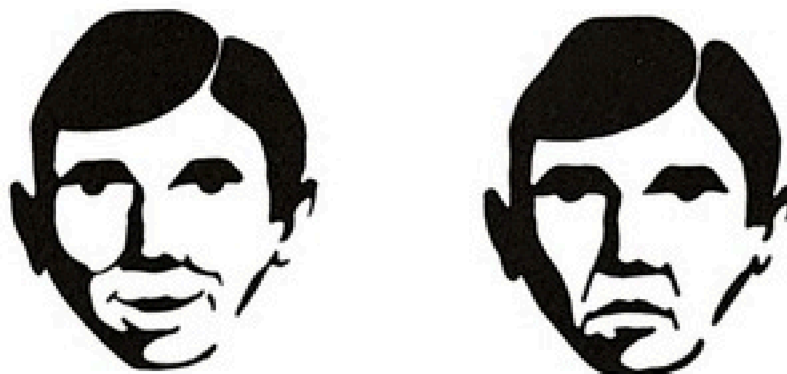
2. Educatieve programma's

2.1 Welke letter is anders?

Dit programma is speciaal ontwikkeld voor kleine kinderen, die nog niet op de basisschool zitten. Er worden negen letters op het beeldscherm gezet, waarbij één letter afwijkt van de overige acht. Het is de bedoeling dat de afwijkende letter wordt herkend en in het programma wordt ingevoerd. Afhankelijk van het antwoord geeft de computer 'goed' of 'fout'. Het programma houdt bij, hoeveel goede antwoorden er gegeven zijn.
Het zal duidelijk zijn dat de kleuter enige uitleg nodig heeft, voordat hij met het programma overweg kan.
Dit programma leent zich uitstekend om een aantal wijzigingen aan te brengen om het zodoende aan de eigen computer aan te passen.
Bij de invoer van het antwoord wordt gebruik gemaakt van het INPUT-statement. Dit houdt in dat men altijd na de invoer de RETURN-toets moet indrukken. Het zal blijken dat de kleintjes regelmatig vergeten de RETURN-toets in te drukken, zodat het programma niet verder gaat. Het is daarom raadzaam het INPUT-statement te vervangen door het GET- of INKEY$-statement. Deze statements stellen de gebruiker in staat om een karakter in te voeren zonder dat de RETURN-toets wordt ingedrukt. De invoer bij dit programma is slechts één karakter, dus men kan het statement gebruiken. Beschikt men over het INKEY$-statement dan moet de invoermodule '*** INVOER I ***' als volgt gewijzigd worden:

```
440 D$ = INKEY$
445 IF D$ = "" THEN 440
```

Een ander idee om het programma te verbeteren, is de manier waarop aan de kleuter wordt kenbaar gemaakt dat het gegeven antwoord goed of fout is. Een aantal mogelijkheden is voorhanden. Er bestaat op verschillende computers de mogelijkheid een melodietje te produceren. Laat men een vrolijk melodietje horen dan was het antwoord goed. Een treurig melodietje geeft aan dat het antwoord fout was. Heeft men een kleurencomputer dan kan men ook met kleuren aangeven dat iets goed of fout is. Kan men deze twee suggesties niet realiseren dan kan men proberen om een gezichtje op het beeldscherm te zetten. Lacht het gezichtje dan was het goed. Kijkt het gezichtje sip dan was het antwoord fout.



Om een idee te geven hoe dit gerealiseerd kan worden, vindt men hier twee subroutines, die men zelf nog in de programma's kan verwerken. De plaats in het programma bepaalt men zelf. Men doet er het beste aan om met ON ... GOSUB-statements deze routines aan te roepen.

```
9000 :
9001 REM *** LACHEND ***
9002 PRINT " /WWWWWWW\"
9003 PRINT " !       !"
9004 PRINT "<< O   O >>"
9005 PRINT "  \  I  /"
9006 PRINT "   \---/  (GOED)"
9007 PRINT "     -"
9008 RETURN
9009 :
9010 REM *** TREURIG ***
9011 PRINT " /WWWWWWW\"
9012 PRINT " !   ?   !"
9013 PRINT "<< O   O >>"
9014 PRINT " !   I   !"
9015 PRINT " ! /---\ ! (FOUT !)"
9016 PRINT "  \     /"
9017 RETURN
9018 :
```

```
 /WWWWWWW\            /WWWWWWW\
 !       !            !   ?   !
<< O   O >>          <<.O   O,>>
  \  I  /             !   I   !
   \---/  (GOED)      ! /---\ !  (FOUT !)
     -                 \     /
```

Programma

```
10 REM *** WELKE LETTER IS ANDERS ***
20 RANDOM                :REM DIT STATEMENT MAG VERVALLEN
30 GOTO 650
40 :
50 REM *** INITIALISEREN ***
60 G = 0                 :REM GOEDE ANTWOORDEN TELLER
70 F = 10                :REM TOTAAL AANTAL VRAGEN
80 T = 0
90 A$ = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
100 RETURN
110 :
120 REM *** NIEUWE VRAAG ***
130 A = INT(RND(0) * 26 + 1)
140 B = INT(RND(0) * 26 + 1)
150 IF A = B THEN 130
160 B$ = LEFT$(A$,A)
170 B$ = RIGHT$(B$,1)
180 C$ = LEFT$(A$,B)
190 C$ = RIGHT$(C$,1)
200 P = INT(RND(0) * 9 + 1)
210 T = T + 1
```

```
220 RETURN
230 :
240 REM *** UITVOER ***
250 CLS                          :REM SCHERM WISSEN
260 PRINT "*** WELKE LETTER IS ANDERS ***"
270 PRINT "AANTAL OPGAVEN...........:";T
280 PRINT "AANTAL GOEDE ANTWOORDEN..:";G
290 PRINT
300 FOR I = 0 TO 2
310 PRINT TAB(10);
320 FOR J = 1 TO 3
330 D$ = B$
340 IF I * 3 + J <> P THEN 360
350 D$ = C$
360 PRINT D$;
370 NEXT J
380 PRINT
390 NEXT I
400 RETURN
410 :
420 REM *** INVOER I ***
430 PRINT
440 PRINT "WELKE LETTER IS ANDERS ";
450 INPUT D$
460 E$ = "NEE "
470 H = 0
480 IF D$ <> C$ THEN 520
490 H = 1
500 E$ = "JA "
510 G = G + 1
520 PRINT E$; "HOOR !  HET WAS ";C$
530 REM ** WACHTLUS **
540 FOR Y = 1 TO 250 * (2 - H)
550 NEXT Y
560 RETURN
570 :
580 REM *** RESULTAAT ***
590 A = INT((G / T ) * 1000 + .5) / 10
600 PRINT A; "PROCENT GOEDE ANTWOORDEN."
610 RETURN
620 :
630 REM **** STUURMODULE ****
640 REM *** INITIALISEREN ***
650 GOSUB 60
660 FOR X = 1 TO F
670 REM *** NIEUWE VRAAG ***
680 GOSUB 130
690 REM *** UITVOER ***
700 GOSUB 250
710 REM *** INVOER I ***
720 GOSUB 430
730 NEXT X
740 REM *** RESULTAAT ***
750 GOSUB 250
760 GOSUB 590
770 END
```

Voorbeeld

```
*** WELKE LETTER IS ANDERS ***
AANTAL OPGAVEN...........: 1
AANTAL GOEDE ANTWOORDEN..: 0

          KKK
          KTK
          KKK

WELKE LETTER IS ANDERS T
JA HOOR !  HET WAS T


*** WELKE LETTER IS ANDERS ***
AANTAL OPGAVEN...........: 2
AANTAL GOEDE ANTWOORDEN..: 1

          BWW
          WWW
          WWW

WELKE LETTER IS ANDERS B
JA HOOR !  HET WAS B


*** WELKE LETTER IS ANDERS ***
AANTAL OPGAVEN...........: 3
AANTAL GOEDE ANTWOORDEN..: 2

          EEE
          XEE
          EEE

WELKE LETTER IS ANDERS X
JA HOOR !  HET WAS X



*** WELKE LETTER IS ANDERS ***
AANTAL OPGAVEN...........: 4
AANTAL GOEDE ANTWOORDEN..: 3

          WWW
          WWW
          JWW

WELKE LETTER IS ANDERS J
JA HOOR !  HET WAS J
```

2.2 Zet in alfabetische volgorde

Op het beeldscherm worden zeven genummerde vakken gevuld met één, twee, of drie letters al naar gelang het niveau waarop men zich bevindt. Deze letters staan in willekeurige volgorde. Men moet trachten door middel van het wisselen van vakken, de letters in alfabetische volgorde te krijgen. Men geeft bij de vraag 'Wissel vak' het nummer van het vak dat men wisselt met het vak, dat men bij de volgende vraag opgeeft.
Denkt men alle letters op volgorde te hebben staan, dan geeft men bij de vraag 'Wissel vak' een nul. Het programma gaat na of alle vakken inderdaad in alfabetische volgorde staan. Wanneer het antwoord goed is, zal een nieuwe reeks in de vakken worden geplaatst. Was het antwoord fout, dan zal de goede volgorde onder de foute volgorde worden afgedrukt.



Een nadeel in het programma is dat iedere keer het hele beeldscherm gewist en daarna weer opgebouwd wordt. Dit komt, omdat het boek in standaard BASIC geschreven is. Hierin komen PEEK- en POKE-opdrachten niet voor. Dat is ook begrijpelijk, omdat deze opdrachten geheugenlokaties rechtstreeks benaderen en omdat elk merk computer anders intern georganiseerd is, kunnen deze opdrachten dus niet gebruikt worden. Voor eigen gebruik kan men natuurlijk wel de programmatuur aanpassen. In dit geval lijkt het raadzaam. Wanneer men met POKE-opdrachten de wijzigingen in het beeldscherm aanbrengt, wordt het plaatje veel rustiger.

Programma

```
10 REM *** ZET IN ALFABETISCHE VOLGORDE ***
20 DIM A$(7), B$(7)
30 RANDOM          :REM DIT STATEMENT KAN WORDEN WEGGELATEN
40 GOTO 1140
50 :
60 REM *** INITIALISEREN ***
70 A$ = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
80 X = (N + 1)/2
90 FOR I = 1 TO 7
100 A$(I) = ""
110 IF (N + 1)/2 - INT((N + 1)/2) = 0 THEN 140
120 Y = (N + 2)/2
130 X = INT(RND(0) * Y + 1)
140 FOR J = 1 TO X
150 A = INT(RND(0) * 26 + 1)
160 B$ = LEFT$(A$,A)
170 A$(I) = A$(I) + RIGHT$(B$,1)
180 NEXT J
190 IF I = 1 THEN 270
200 H = 0                     :REM STUURVARIABELE
210 FOR J = 1 TO I - 1
220 IF A$(I) <> A$(J) THEN 250
230 H = 1
240 J = I
250 NEXT J
260 IF H = 1 THEN 100
270 B$(I) = A$(I)
280 NEXT I
290 REM ** MAT B$ SORTEREN **
300 FOR I = 7 TO 2 STEP -1
310 FOR J = 1 TO I
320 IF B$(I) > B$(J) THEN 360
330 A$ = B$(I)
340 B$(I) = B$(J)
350 B$(J) = A$
360 NEXT J
370 NEXT I
380 B = 1                     :REM BEURT-TELLER
390 RETURN
400 :
410 REM *** UITVOER ***
420 CLS                       :REM SCHERM WISSEN
430 PRINT "*** ZET IN ALFABETISCHE VOLGORDE ***"
440 PRINT "NIVEAU :";N
450 PRINT "BEURT  :";B
460 PRINT
470 PRINT "VAK ";
480 FOR I = 1 TO 7
490 X = 8 + I * 4 - 3    :REM TAB-POSITIE
500 PRINT TAB(X) I;
510 NEXT I
520 PRINT
530 PRINT TAB(8) "!---!---!---!---!---!---!---!"
```

```
540 PRINT "LETTERS";
550 FOR I = 1 TO 7
560 X = 8 + I * 4 - 3    :REM TAB-POSITIE
570 PRINT TAB(X) A$(I);
580 NEXT I
590 PRINT
600 PRINT TAB(8) "!---!---!---!---!---!---!---!"
610 RETURN
620 :
630 REM *** INVOER ***
640 PRINT "KLAAR = 0 "
650 PRINT "WISSEL VAK  :";
660 INPUT X
670 IF X = 0 THEN 810
680 X = INT(X)
690 IF X < 1 OR X > 7 THEN 650
700 PRINT "MET VAK      :";
710 INPUT Y
720 IF Y = 0 THEN 810
730 Y = INT(Y)
740 IF Y < 1 OR Y > 7 THEN 700
750 REM ** VERWERKING **
760 B = B + 1               :REM BEURTEN-TELLER
770 REM * WISSELEN *
780 A$ = A$(X)
790 A$(X) = A$(Y)
800 A$(Y) = A$
810 RETURN
820 :
830 REM *** CONTROLE ***
840 H = 1
850 A$ = "GOED"
860 FOR I = 2 TO 7
870 IF A$(I-1) < A$(I) THEN 910
880 A$ = "FOUT"
890 H = 0
900 I = 7
910 NEXT I
920 RETURN
930 :
940 REM *** RESULTAAT ***
950 GOSUB 420
960 IF H = 1 THEN 1050
970 N = N - 1
980 IF N => 1 THEN 990
990 FOR I = 1 TO 7
1000 X = 8 + I * 4 - 3
1010 PRINT TAB(X) B$(I);
1020 NEXT I
1030 PRINT
1040 PRINT TAB(10) "MOET HET ZIJN  !"
1050 PRINT "HET ANTWOORD IS ";A$;" !"
1060 IF H <> 1 THEN 1110
1070 PRINT "IN ";B;"BEURTEN."
1080 N = N + 1
```

```
1090 IF N < 6 THEN 1110
1100 N = 5
1110 RETURN
1120 :
1130 REM **** STUURMODULE ****
1140 N = 1                    :REM STARTNIVEAU
1150 REM *** INITIALISEREN ***
1160 GOSUB 70
1170 REM *** UITVOER ***
1180 GOSUB 420
1190 REM *** INVOER ***
1200 GOSUB 640
1210 IF X <> 0 AND Y <> 0 THEN 1180
1220 REM *** CONTROLE ***
1230 GOSUB 840
1240 REM *** RESULTAAT ***
1250 GOSUB 950
1260 PRINT "WIL JE DOORGAAN <JA/NEE> ";
1270 INPUT A$
1280 IF LEFT$(A$,1) = "J"  THEN 1160
1290 END
```

Voorbeeld

```
*** ZET IN ALFABETISCHE VOLGORDE ***
NIVEAU : 1
BEURT  : 1

VAK        1   2   3   4   5   6   7
         !---!---!---!---!---!---!---!
LETTERS    E   P   H   Q   T   N   B
         !---!---!---!---!---!---!---!
KLAAR = 0
WISSEL VAK   : 1
MET VAK      : 7

*** ZET IN ALFABETISCHE VOLGORDE ***
NIVEAU : 1
BEURT  : 2

VAK        1   2   3   4   5   6   7
         !---!---!---!---!---!---!---!
LETTERS    B   P   H   Q   T   N   E
         !---!---!---!---!---!---!---!
KLAAR = 0
WISSEL VAK   : 2
MET VAK      : 7

*** ZET IN ALFABETISCHE VOLGORDE ***
NIVEAU : 1
BEURT  : 5

VAK        1   2   3   4   5   6   7
         !---!---!---!---!---!---!---!
LETTERS    B   E   H   P   N   T   Q
         !---!---!---!---!---!---!---!
           B   E   H   N   P   Q   T
            MOET HET ZIJN    !
HET ANTWOORD IS FOUT !
WIL JE DOORGAAN <JA/NEE> NEE
```

2.3 Ontleden

Op het beeldscherm verschijnt een zin. Een woord of gedeelte van de zin wordt onderstreept en tegelijkertijd wordt de vraag gesteld het onderstreepte gedeelte te benoemen. Men heeft de keus uit drie antwoorden. Een van de antwoorden is goed!
Het is de bedoeling dat de speler het goede nummer geeft. Hierna gaat het programma na of het gegeven antwoord goed is of niet. Het resultaat wordt op het beeldscherm getoond. In het geval van een foutief antwoord wordt het goede antwoord twee keer zo lang getoond.

Het aantal zinnen opgeslagen in de interne data file is slechts tien (zie regel 270). Elk DATA-statement bestaat uit een zin. Het eerste getal in het DATA-statement geeft aan uit hoeveel delen de zin bestaat. Er wordt bewust gesproken over delen en niet over woorden, omdat bij benoemen vaak meer woorden betrokken zijn.
Bestaat een deel van de zin uit meer woorden dan moet dit gedeelte ingesloten zijn met aanhalingstekens. Bij een aantal machines dienen alle tekststrings tussen aanhalingstekens te staan! Het laatste karakter uit een deel is altijd een cijfer tussen 1 en 6.
De betekenis van de cijfers is als volgt:

1 = onderwerp
2 = gezegde
3 = meewerkend voorwerp
4 = lijdend voorwerp
5 = bijwoordelijke bepaling
6 = bijvoeglijke bepaling

Hieronder volgt een voorbeeld van een zin, die zou kunnen worden toegevoegd.
Stel men neemt de zin:
'Jan eet zijn brood.'
JAN is het onderwerp van de zin. EET is het gezegde en ZIJN BROOD is lijdend voorwerp. Volgens bovenstaande tabel is het codenummer voor het onderwerp 1, gezegde is 2 en lijdend voorwerp is 4. De hele zin bestaat daarbij uit drie delen. De DATA-regel van de zin 'Jan eet zijn brood.' ziet er dan als volgt uit:

regelnr. DATA 3,JAN1,EET2, "ZIJN BROOD4"

Wanneer men regels toevoegt moet men niet vergeten de variabele Z in regel 270 te veranderen. De variabele Z houdt het aantal opgenomen zinnen bij. Het programma zorgt ervoor dat de cijfers achter de delen niet worden afgedrukt. Aan de zin wordt automatisch een punt toegevoegd.
Beschikt men over een kleurencomputer dan kan men het programma zodanig wijzigen dat het te ontleden gedeelte een andere kleur krijgt. Wanneer men geen kleurencomputer heeft maar met een 'reversed field' kan werken, zou men dít kunnen toepassen.

Programma

```
10 REM *** ONTLEDEN ***
20 RANDOM                        :REM DIT STATEMENT MAG VERVALLEN
30 DIM A$(15), B$(6), D(3)
40 DATA 3, "DE HOND1",LOOPT2,"OP STRAAT5"
50 DATA 4, IK1,GEEF2,JOU3, "EEN BRIEF4"
60 DATA 3, "DE POES1", LIKT2, "ZIJN VACHT4"
70 DATA 3, HET1, REGENT2, "DE HELE DAG5"
80 DATA 3, "DE BLOEMEN1",LIGGEN2, "OP STRAAT5"
90 DATA 3, "DE DOOS1", ZIT2, "VOL ROMMEL4"
100 DATA 4, IK1, GEEF2, "DE HOND3", "EEN KLUIF4"
110 DATA 3, JE1, HEBT2,"EEN LEUKE JURK4"
120 DATA 3, "DE TAS1", STAAT2, "OP TAFEL5"
130 DATA 3, "DIE TAFEL1", STAAT2, "OP ZOLDER5"
140 GOTO 1090
150 :
160 REM *** INITIALISEREN ***
170 B$(1) = "ONDERWERP"
180 B$(2) = "GEZEGDE"
190 B$(3) = "MEEWERKEND VOORWERP"
200 B$(4) = "LIJDEND VOORWERP"
210 B$(5) = "BIJWOORDELIJKE BEPALING"
220 B$(6) = "BIJVOEGLIJKE BEPALING"
230 C$ = "----------------------------------------"
240 T = 0                       :REM TOTAAL AANTAL VRAGEN
250 G = 0                       :REM AANTAL GOED
260 N = 0                       :REM PERC. GOED
270 Z = 10                      :REM AANTAL ZINNEN
280 P = 1                       :REM STUURVARIABELE
290 RETURN
300 :
310 REM *** VRAAG OPHALEN ***
320 X = INT(RND(0) * Z + 1)
330 RESTORE                     :REM DATAPOINTER VOORAAN
340 FOR I = 1 TO X
350 READ A                      :REM AANTAL STRINGS
360 FOR J = 1 TO A
370 READ A$(J)
380 NEXT J
390 NEXT I
400 L = INT(RND(0) * A + 1)
410 C = VAL(RIGHT$(A$(L),1))
420 RETURN
430 :
440 REM ** MULTIPLE CHOICE OPBOUWEN **
450 FOR I = 1 TO 3
460 D(I) = INT(RND(0) * 6 + 1)
470 IF D(I) = C THEN 460
480 IF I = 1 THEN 550
490 H = 0
500 FOR J = 1 TO I - 1
510 IF D(I) <> D(J) THEN 530
520 H = 1
530 NEXT J
```

```
540 IF H = 1 THEN 460
550 NEXT I
560 REM ** GOEDE ANTWOORD PLAATSEN **
570 P = INT(RND(0) * 3 + 1)
580 D(P) = C
590 RETURN
600 :
610 REM *** UITVOER ***
620 CLS                    :REM SCHERM WISSEN
630 PRINT "*** ONTLEDEN ***
640 PRINT
650 PRINT "BEURT                :";T+1
660 PRINT "GOEDE ANTWOORDEN :";G
670 PRINT "PERCENTAGE GOED   :";N
680 PRINT
690 R = 0
700 IF P = 0 THEN 840
710 FOR I = 1 TO A
720 Y = LEN(A$(I)) - 1
730 IF I >= L THEN 750
740 R = R + Y + 1
750 PRINT LEFT$(A$(I),Y);" ";
760 NEXT I
770 PRINT
780 Y = LEN(A$(L)) - 1
790 PRINT TAB(R)LEFT$(C$,Y)
800 PRINT "WAT IS : ";LEFT$(A$(L),Y)
810 FOR I = 1 TO 3
820 PRINT TAB(10) I;" : ";B$(D(I))
830 NEXT I
840 RETURN
850 :
860 REM *** INVOER/ VERWERKING ***
870 PRINT "WAT IS HET GOEDE ANTWOORD <STOP = 0> ";
880 INPUT Q
890 IF Q = 0 THEN 1050
900 IF Q < 0 OR Q > 3 THEN 870
910 H = 0
920 D$ = "=> FOUT <="
930 IF Q <> P THEN 960
940 H = 1
950 D$ = "=> GOED <="
960 PRINT "HET ANTWOORD WAS NR.";P;B$(C)
970 PRINT D$
980 REM ** WACHTLUS **
990 FOR I = 1 TO 250 * (2 - H)
1000 NEXT I
1010 REM ** RESULTAAT VERWERKING **
1020 G = G + H
1030 T = T + 1
1040 N = INT((G/T) * 100 + 0.5)
1050 RETURN
1060 :
1070 REM **** STUURMODULE ****
1080 REM *** INITIALISEREN ***
```

Voorbeeld

```
1090 GOSUB 170
1100 REM *** VRAAG OPHALEN ***
1110 GOSUB 320
1120 REM *** MULTIPLE CHOICE OPBOUWEN ***
1130 GOSUB 450
1140 REM *** UITVOER ***
1150 GOSUB 620
1160 REM *** INVOER/ VERWERKING ***
1170 GOSUB 870
1180 IF Q <> 0 THEN 1110
1190 P = 0
1200 T = T - 1
1210 GOSUB 620
1220 END
```

```
*** ONTLEDEN ***

BEURT              : 1
GOEDE ANTWOORDEN : 0
PERCENTAGE GOED   : 0

JE HEBT EEN LEUKE JURK
--
WAT IS : JE
           1  : GEZEGDE
           2  : ONDERWERP
           3  : MEEWERKEND VOORWERP
WAT IS HET GOEDE ANTWOORD <STOP = 0>  2
HET ANTWOORD WAS NR. 2 ONDERWERP
=> GOED <=


*** ONTLEDEN ***

BEURT              : 2
GOEDE ANTWOORDEN : 1
PERCENTAGE GOED   : 100

HET REGENT DE HELE DAG
           -----------
WAT IS : DE HELE DAG
           1  : GEZEGDE
           2  : BIJWOORDELIJKE BEPALING
           3  : BIJVOEGLIJKE BEPALING
WAT IS HET GOEDE ANTWOORD <STOP = 0>  2
HET ANTWOORD WAS NR. 2 BIJWOORDELIJKE BEPALING
=> GOED <=
```

2.4. Benoemen

Behalve aan (redekundig) ontleden wordt op school ook aan benoemen (taalkundig ontleden) gedaan. Het is de bedoeling werkwoorden, voorzetsels, bezittelijke voornaamwoorden enz. te herkennen. Dit programma is op de zelfde manier opgebouwd als ontleden. Er wordt een zin op het beeldscherm geplaatst en er wordt een woord onderstreept. Aan de hand van een 'multiple choice' wordt gevraagd wat het onderstreepte woord is.
Na invoer wordt het goede antwoord op het beeldscherm gezet en meegedeeld of het ingevoerde antwoord goed was. Mocht blijken dat de tekst in het antwoord te lang of te kort blijft staan, dan kan men dit aanpassen door een wijziging aan te brengen in de wachtlus.
In de interne data-file zijn slechts elf zinnen opgenomen (zie regel 320 in het programma) de opbouw van de DATA-regels is eenvoudig, zodat men zelf dit aantal nog behoorlijk kan uitbreiden. Eerst wordt aangegeven uit hoeveel woorden een zin bestaat. Daarna wordt de zin opgenomen, waarbij elk woord wordt gescheiden van het andere woord door een komma. De punt hoeft niet opgenomen te worden, deze wordt er door het programma zelf achter gezet. Aan elk woord wordt een cijfer vastgeplakt, dat aangeeft wat voor een soort woord het is. De codes vindt men in onderstaande tabel.

codes bij benoemen:
1 werkwoord
2 zelfstandig naamwoord
3 bijvoeglijk naamwoord
4 persoonlijk voornaamwoord
5 lidwoord
6 bijwoord
7 aanwijzend voornaamwoord
8 bezittelijk voornaamwoord
9 voorzetsel

Hieronder volgt een voorbeeld hoe een zin wordt omgezet in een DATA-regel.

'Piet heeft een vlek naast zijn bord.'

De zin bestaat uit zeven woorden. 'Piet' is een zelfstandig naamwoord (2), 'heeft' is een werkwoord (1), 'een' is een lidwoord (5), 'vlek' is een zelfstandig naamwoord (2), 'naast' is een voorzetsel (9), 'zijn' is een bezittelijk voornaamwoord (8) en tenslotte 'bord' is een zelfstandig naamwoord (2). In een DATA-regel ziet dit er als volgt uit:

regelnr DATA 7,PIET2,HEEFT1,EEN5,VLEK2,NAAST9,ZIJN8,BORD2

Men moet niet vergeten regel 320 aan te passen met het aantal zinnen. Het aantal zinnen is slechts beperkt door de geheugenruimte van de gebruikte computer.
Beschikt men over kleurencomputer dan kan men het onderstrepen van het te benoemen woord ook vervangen door het te benoemen woord een andere achtergrondkleur te geven. Bij een computer waarmee men met een reversed field kan werken, dat wil zeggen in plaats van een witte letter op een zwarte achtergrond zwarte letters op een witte achtergrond, kan men op zo'n manier aangeven welk woord benoemd moet worden.

Programma

```
10 REM *** BENOEMEN ***
20 RANDOM                   :REM DIT STATEMENT MAG VERVALLEN
30 DIM A$(15), B$(9), D(3)
40 DATA 5, IK4,GEEF1,DE5,POES2,MELK2
50 DATA 5,EEN5,GROTE3,FIETS2,IS1,NIETS6
60 DATA 5,DIE7,JONGEN2,DRAAGT1,ZIJN8,TAS2
70 DATA 6,HET5,PAARD2,STAAT1,IN9,DE5,STAL2
80 DATA 5,EEN5,GROENE3,PLANT2,BLOEIT1,NOOIT6
90 DATA 6,DEZE7,HOND2,HEEFT1,EEN5,MOOIE3,STAART2
100 DATA 6,ZIJN8,BOEK2,STAAT1,IN9,DE5,KAST2
110 DATA 5,EEN5,PLAATS2,IN9,DE5,SCHOUWBURG2
120 DATA 4,NERGENS6,STAAT1,DE5,WEG2
130 DATA 6,ZIJN8,MOEDER2,KENT1,HET5,VERHAAL2,OOK6
140 DATA 9,DE5,LEZER2,VAN9,DE5,KRANT2,MOET1,DE5
150 DATA PUZZEL2,RADEN1
160 GOTO 1140
170 :
180 REM *** INITIALISEREN ***
190 B$(1) = "WERKWOORD"
200 B$(2) = "ZELFSTANDIG NAAMWOORD"
210 B$(3) = "BIJVOEGLIJK NAAMWOORD"
220 B$(4) = "PERSOONLIJK VOORNAAMWOORD"
230 B$(5) = "LIDWOORD"
240 B$(6) = "BIJWOORD"
250 B$(7) = "AANWIJZEND VOORNAAMWOORD"
260 B$(8) = "BEZITTELIJK VOORNAAMWOORD"
270 B$(9) = "VOORZETSEL"
280 C$ = "------------------------------"
290 T = 0                   :REM TOTAAL AANTAL VRAGEN
300 G = 0                   :REM AANTAL GOED
310 N = 0                   :REM PERC. GOED
320 Z = 11                  :REM AANTAL ZINNEN
330 P = 1                   :REM STUURVARIABELE
340 RETURN
350 :
360 REM *** VRAAG OPHALEN ***
370 X = INT(RND(0) * Z + 1)
380 RESTORE                 :REM DATAPOINTER VOORAAN
390 FOR I = 1 TO X
400 READ A                  :REM AANTAL STRINGS
410 FOR J = 1 TO A
420 READ A$(J)
430 NEXT J
440 NEXT I
450 L = INT(RND(0) * A + 1)
460 C = VAL(RIGHT$(A$(L),1))
470 RETURN
480 :
490 REM ** MULTIPLE CHOICE OPBOUWEN **
500 FOR I = 1 TO 3
510 D(I) = INT(RND(0) * 6 + 1)
520 IF D(I) = C THEN 510
530 IF I = 1 THEN 600
```

```
540 H = 0
550 FOR J = 1 TO I - 1
560 IF D(I) <> D(J) THEN 580
570 H = 1
580 NEXT J
590 IF H = 1 THEN 510
600 NEXT I
610 REM ** GOEDE ANTWOORD PLAATSEN **
620 P = INT(RND(0) * 3 + 1)
630 D(P) = C
640 RETURN
650 :
660 REM *** UITVOER ***
670 CLS                         :REM SCHERM WISSEN
680 PRINT "*** BENOEMEN ***
690 PRINT
700 PRINT "BEURT                :";T+1
710 PRINT "GOEDE ANTWOORDEN :";G
720 PRINT "PERCENTAGE GOED   :";N
730 PRINT
740 R = 0
750 IF P = 0 THEN 890
760 FOR I = 1 TO A
770 Y = LEN(A$(I)) - 1
780 IF I >= L THEN 800
790 R = R + Y + 1
800 PRINT LEFT$(A$(I),Y);" ";
810 NEXT I
820 PRINT "."
830 Y = LEN(A$(L)) - 1
840 PRINT TAB(R)LEFT$(C$,Y)
850 PRINT "WAT IS : ";LEFT$(A$(L),Y)
860 FOR I = 1 TO 3
870 PRINT TAB(10) I;" : ";B$(D(I))
880 NEXT I
890 RETURN
900 :
910 REM *** INVOER/ VERWERKING ***
920 PRINT "WAT IS HET GOEDE ANTWOORD <STOP = 0> ";
930 INPUT Q
940 IF Q = 0 THEN 1100
950 IF Q < 0 OR Q > 3 THEN 920
960 H = 0
970 D$ = "=> FOUT <="
980 IF Q <> P THEN 1010
990 H = 1
1000 D$ = "=> GOED <="
1010 PRINT "HET ANTWOORD WAS NR.";P;B$(C)
1020 PRINT D$
1030 REM ** WACHTLUS **
1040 FOR I = 1 TO 250 * (2 - H)
1050 NEXT I
1060 REM ** RESULTAAT VERWERKING **
1070 G = G + H
1080 T = T + 1
```

```
1090 N = INT((G/T) * 100 + 0.5)
1100 RETURN
1110 :
1120 REM **** STUURMODULE ****
1130 REM *** INITIALISEREN ***
1140 GOSUB 190
1150 REM *** VRAAG OPHALEN ***
1160 GOSUB 370
1170 REM *** MULTIPLE CHOICE OPBOUWEN ***
1180 GOSUB 500
1190 REM *** UITVOER ***
1200 GOSUB 670
1210 REM *** INVOER/ VERWERKING ***
1220 GOSUB 920
1230 IF Q <> 0 THEN 1160
1240 P = 0
1250 T = T - 1
1260 GOSUB 670
1270 END
```

Voorbeeld

```
*** BENOEMEN ***

BEURT             : 1
GOEDE ANTWOORDEN  : 0
PERCENTAGE GOED   : 0

HET PAARD STAAT IN DE STAL .
---
WAT IS : HET
             1  : ZELFSTANDIG NAAMWOORD
             2  : LIDWOORD
             3  : BIJVOEGLIJK NAAMWOORD
WAT IS HET GOEDE ANTWOORD <STOP = 0>  2
HET ANTWOORD WAS NR. 2 LIDWOORD
=> GOED <=


*** BENOEMEN ***

BEURT             : 2
GOEDE ANTWOORDEN  : 1
PERCENTAGE GOED   : 100

DIE JONGEN DRAAGT ZIJN TAS .
                  ----
WAT IS : ZIJN
             1  : PERSOONLIJK VOORNAAMWOORD
             2  : BEZITTELIJK VOORNAAMWOORD
             3  : ZELFSTANDIG NAAMWOORD
WAT IS HET GOEDE ANTWOORD <STOP = 0>  2
HET ANTWOORD WAS NR. 2 BEZITTELIJK VOORNAAMWOORD
=> GOED <=


*** BENOEMEN ***

BEURT             : 5
GOEDE ANTWOORDEN  : 3
PERCENTAGE GOED   : 75

EEN GROENE PLANT BLOEIT NOOIT .
           -----
WAT IS : PLANT
             1  : PERSOONLIJK VOORNAAMWOORD
             2  : ZELFSTANDIG NAAMWOORD
             3  : BIJVOEGLIJK NAAMWOORD
WAT IS HET GOEDE ANTWOORD <STOP = 0>  2
HET ANTWOORD WAS NR. 2 ZELFSTANDIG NAAMWOORD
=> GOED <=
```

2.5 Algemeen leerprogramma

Uit het hoofd leren is vaak een kwestie van constant blijven herhalen. Overhoren is de enige manier om na te gaan of men zijn zaakjes kent. Dit programma is speciaal ontwikkeld om dit stampwerk en het overhoren te vergemakkelijken. Tegelijkertijd heeft het als voordeel dat men ook weet hoe men de zaken schrijft, die overhoord worden.



Het programma stelt door elkaar twee verschillende vragen. In dit geval is de interne data file gevuld met vragen over aardrijkskunde. Het gaat hierbij om de vraag 'Welk land hoort bij...' en 'Wat is de hoofdstad van...'. Het programma selecteert één paar gegevens en één vraag. De gebruiker wordt verzocht het antwoord op de vraag te geven. Blijkt het antwoord fout te zijn, dan onthoudt het programma bij welk paar gegevens een fout is gemaakt. Wordt er een nieuwe vraag gesteld, dan zal de computer geneigd zijn vragen te stellen waarop eerder een foutief antwoord was verstrekt. In totaal onthoudt de computer 10 foutieve antwoorden. Wordt een vraag, die eerst fout beantwoord was, nu goed beantwoord, dan wordt deze verwijderd uit de foutlijst.
Het is maar toevallig dat de interne data-file gevuld is met aardrijkskunde. Hij zou ook gevuld kunnen zijn met Engelse woordjes. Hiertoe moet men de inhoud van de interne data-file wijzigen. Men kopieert het programma en wist de regels 20 t/m 90. Men begint met in regel 20 op te nemen wat het onderwerp is. In dit geval Engels. In regel 30 neemt men de eerste vraag op. Bijvoorbeeld: 'Wat is het Engelse woord voor'. Op regel 40 neemt

men de tweede vraag op 'wat is het Nederlandse woord voor'. In DATA-statements gaat dit er als volgt uitzien:

20 DATA 'ENGELS'
30 DATA 'WAT IS HET ENGELSE WOORD VOOR'
40 DATA 'WAT IS HET NEDERLANDSE WOORD VOOR'

Hierna moet men de combinatie van woorden gaan opnemen. In dit programma kunnen dit maximaal 20 paren zijn. Het laatste paar moet <STOP>,<STOP> zijn en moet ook zo worden ingetikt.

50 DATA GARDEN,TUIN, HOUSE,HUIS, AIRPLANE,VLIEGTUIG
60 DATA ......
90 DATA <STOP>,<STOP>

Men moet er op letten dat men regel 90 niet vergeet. Deze regel zorgt ervoor dat het programma weet dat het aan het einde is gekomen van de interne data-file. Vergeet men <STOP>,<STOP> in de laatste DATA-regel te zetten dan zal het programma stuklopen. Wanneer men vindt dat men te weinig vragen kan stellen, kan de dimensie van matrix A$ worden vergroot. De matrix staat in het programma gedimensioneerd op 20 bij 2. Men kan dus maximaal 20 paren gegevens opnemen. Wil men meer paren gegevens opnemen dan zal men het aantal regels in matrix A$ moeten uitbreiden. Wanneer men in regel 20 van het programma A$ (20,2) wijzigt in A$ (40,2) dan kan men 40 paren Engelse woorden opnemen.

Programma

```
10 REM *** ALGEMEEN LEERPROGRAMMA ***
20 DATA AARDRIJKSKUNDE
30 DATA "WAT IS DE HOOFDSTAD VAN"
40 DATA "WELK LAND HOORT BIJ"
50 DATA AMSTERDAM,NEDERLAND,BRUSSEL,BELGIE
60 DATA LONDEN,ENGELAND,BONN,DUITSLAND,WARSCHAU,POLEN
70 DATA WENEN,OOSTENRIJK,BERN,ZWITSERLAND
80 DATA PARIJS,FRANKRIJK,ROME,ITALIE
90 DATA <STOP>,<STOP>
100 DIM A$(20,2), B$(2), C(10)
110 RANDOM                  :REM DIT STATEMENT MAG VERVALLEN
120 GOTO 1150
130 :
140 REM *** LEZEN INTERNE DATAFILE ***
150 RESTORE
160 READ T$, B$(1), B$(2)
170 X = 0                   :REM HULPTELLER
180 X = X + 1
190 READ A$(X,1), A$(X,2)
200 IF A$(X,1) = "<STOP>"THEN 230
210 N = X                   :REM OPGAVENTELLER
220 IF N < 20 THEN 180
230 RETURN
240 :
250 REM *** INITIALISEREN ***
260 REM ** FOUTMATRIX SCHOONVEGEN **
```

```
270 FOR I = 1 TO 10
280 C(I) = 0
290 NEXT I
300 REM ** VARIABELEN OP STARTWAARDE **
310 T = 0                   :REM TOTAAL VRAGEN
320 G = 0                   :REM AANTAL GOEDEN
330 F = 0                   :REM FOUTEN IN FOUTTABEL
340 Y = 0                   :REM GOEDPERC.
350 RETURN
360 :
370 REM *** VRAAG OPHALEN ***
380 J = INT(RND(0) * 2 + 1)
390 H = INT(RND(0) * N + 1)
400 IF RND(0) > 0.6 OR F = 0 THEN 430
410 H = INT(RND(0) * F + 1)
420 H = C(H)
430 IF T = 0 THEN 450
440 Y = INT((G/T)*1000 + 0.5)/10
450 Q$ = A$(H,1)
460 Z$ = A$(H,2)
470 IF J = 1 THEN 500
480 Q$ = A$(H,2)
490 Z$ = A$(H,1)
500 RETURN
510 :
520 REM *** UITVOER ***
530 CLS                     :REM SCHERM WISSEN
540 PRINT "*** ALGEMEEN LEERPROGRAMMA ***"
550 PRINT "ONDERWERP   :";T$
560 PRINT "BEURT       :";T + 1
570 PRINT "PERC. GOED :";Y
580 PRINT
590 PRINT "STOPPEN IS : 'STOP'+<RETURN>"
600 PRINT B$(J);" ";Z$;" ";
610 RETURN
620 :
630 REM *** INVOER/VERWERKING ***
640 INPUT E$
650 Z = 3
660 IF E$ = "STOP" THEN 790
670 T = T + 1               :REM BEURT + 1
680 Z = 0
690 IF E$ <> Q$ THEN 710
700 Z = 1
710 G = G + Z
720 ON Z + 1 GOSUB 970  ,820
730 PRINT TAB(10) R$
740 PRINT "HET ANTWOORD WAS :";Q$
750 REM ** WACHTLUS **
760 FOR I = 1 TO 250 * (2 - Z)
770 NEXT I
780 Y = ((G/T) * 1000 + 0.5)/10
790 RETURN
800 :
810 REM *** ANTWOORD WAS GOED ***
```

```
820 R$ = "=> GOED <="
830 IF F = 0 THEN 1010
840 FOR I = 1 TO F
850 IF H <> C(I) THEN 930
860 FOR J = I TO F - 1
870 C(J) = C(J + 1)
880 NEXT J
890 I = F                    :REM LUS OP EINDWAARDE
900 F = F - 1
910 IF F >= 0 THEN 930
920 F = 0
930 NEXT I
940 RETURN
950 :
960 REM *** ANTWOORD WAS FOUT ***
970 R$ = "=> FOUT <="
980 IF F > 9 THEN 1010
990 F = F + 1
1000 C(F) = H
1010 RETURN
1020 :
1030 REM *** RESULTAAT ***
1040 CLS
1050 PRINT "*** ALGEMEEN LEERPROGRAMMA ***"
1060 PRINT "ONDERWERP               :";T$
1070 PRINT
1080 PRINT "TOTAAL AANTAL BEURTEN  :";T
1090 PRINT "TOTAAL GOED BEANTWOORD :";G
1100 PRINT "PERCENTAGE GOED        :";Y
1110 RETURN
1120 :
1130 REM **** STUURMODULE ****
1140 REM *** INTERNE FILE LEZEN ***
1150 GOSUB 150
1160 REM *** INITIALISEREN ***
1170 GOSUB 270
1180 REM *** VRAAG OPHALEN ***
1190 GOSUB 380
1200 REM *** UITVOER ***
1210 GOSUB 530
1220 REM *** INVOER/VERWERKING ***
1230 GOSUB 640
1240 IF Z < 3 THEN 1190
1250 REM *** RESULTAAT ***
1260 GOSUB 1040
1270 END
```

Voorbeeld

```
*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP   :AARDRIJKSKUNDE
BEURT       : 1
PERC. GOED : 0
```

```
STOPPEN IS : 'STOP'+<RETURN>
WAT IS DE HOOFDSTAD VAN ITALIE ROME
          => GOED <=
HET ANTWOORD WAS :ROME


*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP   :AARDRIJKSKUNDE
BEURT       : 2
PERC. GOED : 100

STOPPEN IS : 'STOP'+<RETURN>
WELK LAND HOORT BIJ BERN ZWITSERLAND
          => GOED <=
HET ANTWOORD WAS :ZWITSERLAND


*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP   :AARDRIJKSKUNDE
BEURT       : 3
PERC. GOED : 100

STOPPEN IS : 'STOP'+<RETURN>
WELK LAND HOORT BIJ WENEN OOSTENRIJK
          => GOED <=
HET ANTWOORD WAS :OOSTENRIJK


*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP   :AARDRIJKSKUNDE
BEURT       : 4
PERC. GOED : 100

STOPPEN IS : 'STOP'+<RETURN>
WELK LAND HOORT BIJ WARSCHAU POLEN
          => GOED <=
HET ANTWOORD WAS :POLEN


*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP   :AARDRIJKSKUNDE
BEURT       : 5
PERC. GOED : 100

STOPPEN IS : 'STOP'+<RETURN>
WELK LAND HOORT BIJ BRUSSEL STOP


*** ALGEMEEN LEERPROGRAMMA ***
ONDERWERP                 :AARDRIJKSKUNDE

TOTAAL AANTAL BEURTEN   : 4
TOTAAL GOED BEANTWOORD : 4
PERCENTAGE GOED         : 100
```

2.6 Welk cijfer is anders?

Evenals het programma 'Welke letter is anders?', is ook dit programma ontwikkeld voor kleuters. Op het beeldscherm worden negen cijfers gezet. Acht cijfers zijn hetzelfde, één cijfer wijkt af. Ook hier is het de bedoeling dat het afwijkende cijfer wordt herkend. Het afwijkende cijfer wordt ingevoerd in het programma en het programma gaat na of het antwoord goed was. Er is voor gezorgd dat het te herkennen cijfer niet steeds op dezelfde plaats staat. Dat zou het programma al te gemakkelijk maken. Men moet de kleuter wel goed uitleggen wat de bedoeling is van het programma.
Ook bij dit programma wordt de score bijgehouden. Veranderingen in het programma zijn eenvoudig aan te brengen. Een aantal suggesties is reeds gegeven bij het programma 'Welke letter is anders'. Heeft men een kleurencomputer dan kan men het vak van het afwijkende cijfer na het antwoord een andere kleur geven en dan het afwijkende cijfer ernaast in tweemaal de grootte. Beschikt men over een computer waarbij men de achtergrond van een karakter kan wisselen van zwart naar wit (reversed field) dan kan men op die manier de plaats van het afwijkende cijfer na het antwoord aangeven.
Na iedere vraag wordt het gehele plaatje opnieuw opgebouwd. Het zou aanbeveling verdienen de gegevens in het plaatje te POKEn. Het geheel wordt er wel een stuk rustiger door.

Programma

```
10 REM *** WELK CIJFER IS ANDERS ***
20 RANDOM                    :REM DIT STATEMENT MAG VERVALLEN
30 GOTO 630
40 :
50 REM *** INITIALISEREN ***
60 G = 0                     :REM GOEDE ANTWOORDEN TELLER
70 T = 10                    :REM TOTAAL AANTAL VRAGEN
80 A$ = "1234567890"
90 RETURN
100 :
110 REM *** NIEUWE VRAAG ***
120 A = INT(RND(0) * 10 + 1)
130 B = INT(RND(0) * 10 + 1)
140 IF A = B THEN 120
150 B$ = LEFT$(A$,A)
160 B$ = RIGHT$(B$,1)
170 C$ = LEFT$(A$,B)
180 C$ = RIGHT$(C$,1)
190 P = INT(RND(0) * 9 + 1)
200 RETURN
210 :
220 REM *** UITVOER ***
230 CLS                            :REM SCHERM WISSEN
240 PRINT "*** WELK CIJFER IS ANDERS ***"
250 PRINT "AANTAL OPGAVEN...........:";X
260 PRINT "AANTAL GOEDE ANTWOORDEN..:";G
270 PRINT
280 FOR I = 0 TO 2
290 PRINT TAB(10);
300 FOR J = 1 TO 3
```

```
310 D$ = B$
320 IF I * 3 + J <> P THEN 340
330 D$ = C$
340 PRINT D$;
350 NEXT J
360 PRINT
370 NEXT I
380 RETURN
390 :
400 REM *** INVOER ***
410 PRINT
420 PRINT "WELK CIJFER IS ANDERS ";
430 INPUT D$
440 E$ = "NEE "
450 IF D$ <> C$ THEN 500
460 H = 0
470 E$ = "JA "
480 H = 1
490 G = G + 1
500 PRINT E$; "HOOR !  HET WAS ";C$
510 REM ** WACHTLUS **
520 FOR Y = 1 TO 250 * (2 - H)
530 NEXT Y
540 RETURN
550 :
560 REM *** RESULTAAT ***
570 A = INT((G / T ) * 1000 + 0.5) / 10
580 PRINT A; "PROCENT GOEDE ANTWOORDEN."
590 RETURN
600 :
610 REM **** STUURMODULE ****
620 REM *** INITIALISEREN ***
630 GOSUB 60
640 FOR X = 1 TO T
650 REM *** NIEUWE VRAAG ***
660 GOSUB 120
670 REM *** UITVOER ***
680 GOSUB 230
690 REM *** INVOER ***
700 GOSUB 410
710 NEXT X
720 REM *** RESULTAAT ***
730 X = 10
740 GOSUB 230
750 GOSUB 570
760 END
```

Voorbeeld

```
*** WELK CIJFER IS ANDERS ***
AANTAL OPGAVEN...........: 1
AANTAL GOEDE ANTWOORDEN..: 0

          444
          447
          444
WELK CIJFER IS ANDERS 7
JA HOOR !  HET WAS 7


*** WELK CIJFER IS ANDERS ***
AANTAL OPGAVEN...........: 2
AANTAL GOEDE ANTWOORDEN..: 1

          611
          111
          111
WELK CIJFER IS ANDERS 6
JA HOOR !  HET WAS 6


*** WELK CIJFER IS ANDERS ***
AANTAL OPGAVEN...........: 3
AANTAL GOEDE ANTWOORDEN..: 2

          818
          888
          888
WELK CIJFER IS ANDERS 1
JA HOOR !  HET WAS 1
```

2.7 Welk getal ontbreekt?

Dit programma is gebaseerd op het honderdveld, een blad met de getallen van één tot en met honderd op een speciale manier gerangschikt. Het programma werkt met automatische niveaus. Dit heeft als voordeel dat de kleintjes ermee kunnen werken (niveau 1), maar ook degenen die al in de tweede klas van de basisschool zitten.
Het programma zet een aangesloten rij getallen op het beeldscherm en laat daar een getal van weg. Het gaat er om dit ontbrekende getal in te vullen. Blijkt het antwoord juist te zijn, dan wordt de niveau-indicator met één verhoogd. Was het fout dan wordt de indicator met één verlaagd. Het niveau zelf is gelijk aan veelvouden van vijf. Tegelijkertijd bepaalt het niveau ook wat voor soort getallenrij er gegenereerd zal worden. Geeft men als antwoord een nul (0) dan geeft men daarmee te kennen dat men het programma wenst te verlaten. Tenslotte wordt de uiteindelijke score nog op het beeldscherm gezet. Er wordt aangegeven hoeveel vragen er gesteld zijn en hoeveel vragen goed zijn beantwoord. Tevens wordt het percentage goed bepaald.
Beschikt men over een kleurencomputer dan is dit programma bijzonder geschikt om veranderingen aan te brengen. Het programma geeft met een vraagteken de plaats aan, waar een getal ontbreekt. Dit blijkt op sommige computers vrij onduidelijk te zijn. Men kan nu voor de duidelijkheid het vak, waar het getal ontbreekt ook nog een andere kleur geven. Heeft men een computer, waarbij een karakter kan worden omgedraaid (reversed field), zoals de Commodore dan kan men het op die manier aangeven.

Programma

```
10 REM *** WELK GETAL ONTBREEKT ***
20 DIM A(9)
30 RANDOM                  :REM DIT STATEMENT MAG VERVALLEN
40 GOTO 860
50 :
60 REM *** INITIALISEREN ***
70 T = 0                   :REM AANTAL VRAGEN
80 G = 0                   :REM AANTAL GOEDEN
90 N = 1                   :REM NIVEAU
100 P = 1                  :REM NIVEAUINDICATIE
110 K = 0                  :REM PERC. GOEDEN
120 A$ = "!---!---!---!"
130 RETURN
140 :
150 REM *** VRAAG OPBOUWEN ***
160 X = INT(RND(0) * (N - 1) * 10)
170 FOR I = 1 TO 9
180 A(I) = I + X
190 NEXT I
200 X = INT(RND(0) * 9 + 1)
210 A = A(X)
220 A(X) = 0
230 IF T = 0 THEN 250
240 K = INT((G/T) * 1000 + 0.5)/10
250 RETURN
260 :
270 REM *** UITVOER ***
280 CLS                    :REM SCHERM WISSEN
```

```
290 PRINT "*** WELK GETAL ONTBREEKT ***"
300 PRINT "NIVEAU          :";N
310 PRINT "BEURT           :";T + 1
320 PRINT "PERC. GOEDEN    :";K
330 PRINT
340 PRINT TAB(5) A$
350 FOR I = 0 TO 2
360 FOR J = 1 TO 3
370 Y = I * 3 + J
380 H = J * 4 - 3 + 5
390 IF A(Y) <> 0 THEN 420
400 PRINT TAB(H)" ?";
410 GOTO 430
420 PRINT TAB(H)A(Y);
430 NEXT J
440 PRINT
450 PRINT TAB(5)A$
460 NEXT I
470 RETURN
480 :
490 REM *** INVOER / VERWERKING ***
500 PRINT "WELK GETAL ONTBREEKT <STOP = 0> ";
510 INPUT R
520 IF R = 0 THEN 730
530 T = T + 1              :REM BEURT + 1
540 P = P - 1
550 H = 0                 :REM HULPVARIABELE
560 B$ = "=> FOUT <="
570 IF R <> A THEN 620
580 B$ = "=> GOED <="
590 H = 1
600 G = G + 1
610 P = P + 2
620 IF P < 50 THEN 640
630 P = 50
640 IF P > 0 THEN 660
650 P = 1
660 N = INT(P/5 + 1)            :REM NIVEAU
670 PRINT TAB(10) B$
680 PRINT "HET ANTWOORD WAS "; A
690 REM ** WACHTLUS **
700 FOR I = 1 TO 250 * (2 - H)
710 NEXT I
720 K = INT((G/T) * 1000 + 0.5)/10
730 RETURN
740 :
750 REM *** RESULTAAT ***
760 CLS                         :REM SCHERM WISSEN
770 PRINT "*** WELK GETAL ONTBREEKT ***"
780 PRINT
790 PRINT "BEREIKT NIVEAU     :";N
800 PRINT "AANTAL BEURTEN     :";T
810 PRINT "PERC. GOEDEN       :";K
820 RETURN
830 :
```

```
840 REM **** STUURMODULE ****
850 REM *** INITIALISEREN ***
860 GOSUB 70
870 REM *** VRAAG OPBOUWEN ***
880 GOSUB 160
890 REM *** UITVOER ***
900 GOSUB 280
910 REM *** INVOER / VERWERKING ***
920 GOSUB 500
930 IF R <> 0 THEN 880
940 REM *** RESULTAAT ***
950 GOSUB 760
960 END
```

Voorbeeld

```
*** WELK GETAL ONTBREEKT ***
NIVEAU          : 1
BEURT           : 1
PERC. GOEDEN    : 0

         !---!---!---!
           1   2   3
         !---!---!---!
           4   5   ?
         !---!---!---!
           7   8   9
         !---!---!---!
WELK GETAL ONTBREEKT <STOP = 0>   6
          => GOED <=
HET ANTWOORD WAS  6


*** WELK GETAL ONTBREEKT ***
NIVEAU          : 1
BEURT           : 2
PERC. GOEDEN    : 100

         !---!---!---!
           1   2   3
         !---!---!---!
           4   5   6
         !---!---!---!
           7   8   ?
         !---!---!---!
WELK GETAL ONTBREEKT <STOP = 0>   9
          => GOED <=
HET ANTWOORD WAS  9
```

```
*** WELK GETAL ONTBREEKT ***
NIVEAU          : 1
BEURT           : 3
PERC. GOEDEN    : 100

      !---!---!---!
        1   2   3
      !---!---!---!
        4   5   6
      !---!---!---!
        ?   8   9
      !---!---!---!
WELK GETAL ONTBREEKT <STOP = 0>  7
         => GOED <=
HET ANTWOORD WAS  7


*** WELK GETAL ONTBREEKT ***
NIVEAU          : 1
BEURT           : 4
PERC. GOEDEN    : 100

      !---!---!---!
        1   2   3
      !---!---!---!
        4   5   6
      !---!---!---!
        7   ?   9
      !---!---!---!
WELK GETAL ONTBREEKT <STOP = 0>  8
         => GOED <=
HET ANTWOORD WAS  8


*** WELK GETAL ONTBREEKT ***
NIVEAU          : 2
BEURT           : 6
PERC. GOEDEN    : 100

      !---!---!---!
        8   9   10
      !---!---!---!
        11  ?   13
      !---!---!---!
        14  15  16
      !---!---!---!
WELK GETAL ONTBREEKT <STOP = 0>  0


*** WELK GETAL ONTBREEKT ***

BEREIKT NIVEAU      : 2
AANTAL BEURTEN      : 5
PERC. GOEDEN        : 100
```

2.8 Zet de getallen op volgorde

Qua structuur is dit programma hetzelfde als het programma 'Zet in alfabetische volgorde'. Het enige verschil is dat dit programma met getallen werkt, terwijl het eerder genoemde programma met letters werkt. Het programma zet zeven getallen in een willekeurige volgorde in de vakken. De speler moet proberen deze getallen in volgorde van laag naar hoog te krijgen. Denkt hij er in geslaagd te zijn, dan geeft hij bij de vraag 'Wissel van' een nul. Het programma zal controleren of de rij getallen inderdaad in de goede volgorde staat. Blijkt het antwoord niet te voldoen, dan zal de goede rij onder de foutieve rij geplaatst worden. Men kan dan zien wat men fout heeft gedaan.



Bewust is in dit programma geen niveaumechanisme ingebouwd. Dit wordt aan de gebruiker van het programma overgelaten. Aan de hand van de verschillende programma's in het educatieve deel moet het niet te moeilijk zijn zo'n mechanisme in te bouwen. Men kan beginnen met één- of tweecijferige getallen, daarna tweecijferige getallen en tenslotte driecijferige getallen. Het is niet moeilijk dit niveaumechanisme in te bouwen.
Veel succes!

Programma

```
10 REM *** ZET DE GETALLEN OP VOLGORDE ***
20 DIM A(9), B(9)
30 RANDOM          :REM DIT STATEMENT KAN WORDEN WEGGELATEN
40 GOTO 990
50 :
60 REM *** INITIALISEREN ***
70 FOR I = 1 TO 9
80 A(I) = INT(RND(0) * 20 + 1)
90 IF I = 1 THEN 170
100 H = 0                   :REM STUURVARIABELE
110 FOR J = 1 TO I - 1
120 IF A(I) <> A(J) THEN 150
130 H = 1
140 J = I
150 NEXT J
160 IF H = 1 THEN 80
170 B(I) = A(I)
180 NEXT I
190 REM ** MAT B SORTEREN **
200 FOR I = 9 TO 2 STEP -1
210 FOR J = 1 TO I
220 IF B(I) > B(J) THEN 260
230 H = B(I)
240 B(I) = B(J)
250 B(J) = H
260 NEXT J
270 NEXT I
280 B = 1                     :REM BEURT-TELLER
290 RETURN
300 :
310 REM *** UITVOER ***
320 CLS                     :REM SCHERM WISSEN
330 PRINT "*** ZET DE GETALLEN OP VOLGORDE ***"
340 PRINT "BEURT  :";B
350 PRINT
360 PRINT "VAK ";
370 FOR I = 1 TO 9
380 X = 5 + I * 4 - 3    :REM TAB-POSITIE
390 PRINT TAB(X) I;
400 NEXT I
410 PRINT
420 PRINT TAB(5) "!---!---!---!---!---!---!---!---!---!"
430 PRINT "GETAL ";
440 FOR I = 1 TO 9
450 X = 5 + I * 4 - 3    :REM TAB-POSITIE
460 PRINT TAB(X) A(I);
470 NEXT I
480 PRINT
490 PRINT TAB(5) "!---!---!---!---!---!---!---!---!---!"
500 RETURN
510 :
520 REM *** INVOER ***
530 PRINT "KLAAR = 0 "
```

```
540 PRINT "WISSEL VAK   :";
550 INPUT X
560 IF X = 0 THEN 700
570 X = INT(X)
580 IF X < 1 OR X > 9 THEN 540
590 PRINT "MET VAK      :";
600 INPUT Y
610 IF Y = 0 THEN 700
620 Y = INT(Y)
630 IF Y < 1 OR Y > 9 THEN 590
640 REM ** VERWERKING **
650 B = B + 1              :REM BEURTEN-TELLER
660 REM * WISSELEN *
670 H = A(X)
680 A(X) = A(Y)
690 A(Y) = H
700 RETURN
710 :
720 REM *** CONTROLE ***
730 H = 1
740 A$ = "GOED"
750 FOR I = 2 TO 9
760 IF A(I-1) < A(I) THEN 800
770 A$ = "FOUT"
780 H = 0
790 I = 9
800 NEXT I
810 RETURN
820 :
830 REM *** RESULTAAT ***
840 GOSUB 320
850 IF H = 1 THEN 920
860 FOR I = 1 TO 9
870 X = 5 + I * 4 - 3
880 PRINT TAB(X) B(I);
890 NEXT I
900 PRINT
910 PRINT TAB(10) "MOET HET ZIJN  !"
920 PRINT "HET ANTWOORD IS ";A$;" !"
930 IF H <> 1 THEN 950
940 PRINT "IN ";B;"BEURTEN."
950 RETURN
960 :
970 REM **** STUURMODULE ****
980 REM *** INITIALISEREN ***
990 GOSUB 70
1000 REM *** UITVOER ***
1010 GOSUB 320
1020 REM *** INVOER ***
1030 GOSUB 530
1040 IF X <> 0 AND Y <> 0 THEN 1010
1050 REM *** CONTROLE ***
1060 GOSUB 730
1070 REM *** RESULTAAT ***
1080 GOSUB 840
1090 END
```

Voorbeeld

```
*** ZET DE GETALLEN OP VOLGORDE ***
BEURT  : 1

VAK      1   2   3   4   5   6   7   8   9
       !---!---!---!---!---!---!---!---!---!
GETAL   13   5  10  16   2   3  19  18  20
       !---!---!---!---!---!---!---!---!---!
KLAAR = 0
WISSEL VAK   : 7
MET VAK      : 8

*** ZET DE GETALLEN OP VOLGORDE ***
BEURT  : 2

VAK      1   2   3   4   5   6   7   8   9
       !---!---!---!---!---!---!---!---!---!
GETAL   13   5  10  16   2   3  18  19  20
       !---!---!---!---!---!---!---!---!---!
KLAAR = 0
WISSEL VAK   : 6
MET VAK      : 4

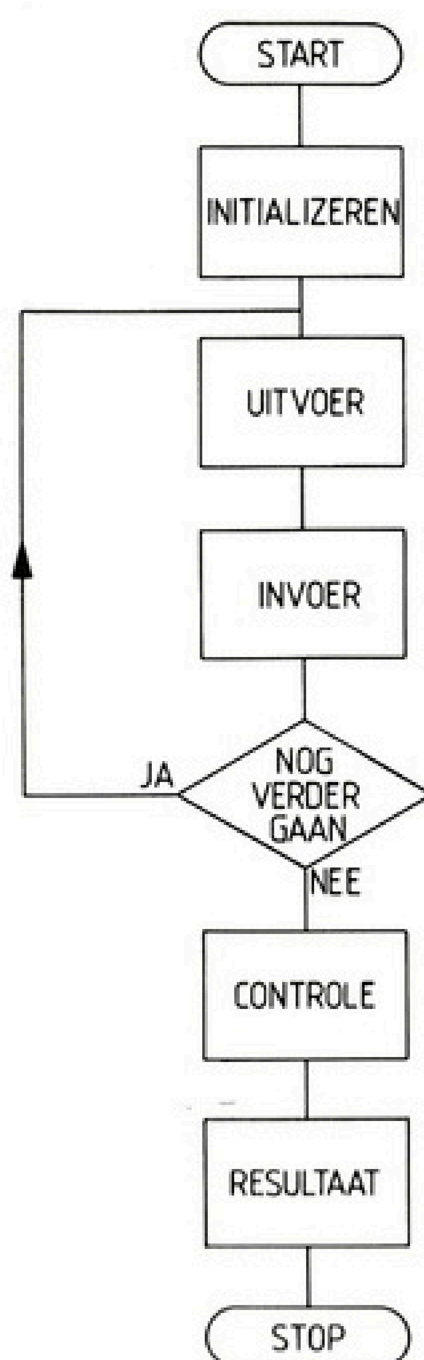

*** ZET DE GETALLEN OP VOLGORDE ***
BEURT  : 6

VAK      1   2   3   4   5   6   7   8   9
       !---!---!---!---!---!---!---!---!---!
GETAL    2   3   5  10  13  16  18  19  20
       !---!---!---!---!---!---!---!---!---!
HET ANTWOORD IS GOED !
IN  6 BEURTEN.
```

Werking van het programma

1. In matrix A worden negen willekeurige getallen geplaatst. Tegelijkertijd worden dezelfde getallen ook in matrix B geplaatst. Tenslotte wordt matrix B op volgorde gesorteerd. Indien bij de controle blijkt dat de ingevoerde rij foutief was, dan kan matrix B worden getoond.
2. De inhoud van matrix A wordt op het beeldscherm gezet.
3. De speler wisselt twee vakken met elkaar of hij geeft een nul ten teken dat hij wil stoppen.
4. Was de invoer ongelijk aan nul, dan terug naar punt 2.
5. De inhoud van matrix A wordt gecontroleerd of deze op oplopende volgorde staat.
6. De resultaten worden op het beeldscherm gezet.
7. Einde.

In een stroomschema ziet het basisidee er als volgt uit:



Wanneer men de stuurmodule uit het programma vergelijkt met het hiervoor geschetste basisidee, dan zal zeker de overeenkomst opvallen.
Toen dit programma ontwikkeld werd, is eerst de stuurmodule gemaakt, daarna de uitvoermodule. Daar zat namelijk het grootste probleem. De programma's in dit boek zijn in

standaard BASIC geschreven en dat houdt in dat de schermopmaak vrij beperkt is. Toen deze module eenmaal klaar was, is de module 'initialiseren' gebouwd. Daarna kon ook de uitvoermodule getest worden. Vervolgens zijn de andere modules geschreven en afzonderlijk getest. Tenslotte zijn de modules aan de stuurmodule gekoppeld.
Wanneer men de stuurmodule van Zeeslag naast de stuurmodule van dit programma legt, zal blijken dat de opzet van beide programma's ongeveer gelijk is. Qua structuur lijken beide programma's op elkaar, hetgeen ook de bedoeling was. Bij de ontwikkeling was het mogelijk razendsnel fouten te lokaliseren en te verhelpen.

2.9 Sommen

In de rij educatieve programma's mag een programma als 'Sommen' niet ontbreken. Dit programma genereert achter elkaar hoofdrekensommen. Op de lagere niveaus zijn het eenvoudige optel- en aftreksommen. De uitkomst komt niet hoger dan 20. Wanneer men goede antwoorden op de sommen geeft, stijgt het niveau zodat ook de moeilijkheidsgraad stijgt. Er komen dan ook vermenigvuldig- en deelsommen. De uitkomst van dit soort sommen zal op het hoogste niveau nooit uitstijgen boven de tafel van 10.
Het programma genereert twee soorten opgaven, De eerste soort opgaven heeft de vorm A + B = ? De tweede soort heeft de vorm A + ? = C. Voor het plusteken kan ook een min-, vermenigvuldig- of deelteken staan. De getallen, die getrokken worden, zijn volkomen willekeurig en de hoogte is alleen afhankelijk van het niveau waarop men zich bevindt.
Op het beeldscherm wordt tevens bijgehouden hoeveel opgaven men gemaakt heeft, hoeveel opgaven goed beantwoord zijn en hoe hoog het percentage goede antwoorden is. Wordt een vraag fout beantwoord, dan wordt het goede antwoord getoond. Mocht blijken dat de wachtlus in regel 1100 te lang of te kort is dan kan men door aanpassing van het getal 250 hierin verandering aanbrengen.
Wil men stoppen met het programma, dan zal men bij het antwoord −9 moeten invoeren. Het programma toont dan het uiteindelijk behaalde resultaat. In het programma komen geen negatieve getallen als antwoord voor. Het zijn uitsluitend positieve getallen zoals men dat op de basisschool gewend is.

Programma

```
10 REM *** SOMMEN ****
20 DIM A$(4)
30 RANDOM           :REM DIT STATEMENT KAN MEN WEGLATEN
40 GOTO 1280
50 :
60 REM *** INITIALISEREN ***
70 A$(1) = "+"
80 A$(2) = "-"
90 A$(3) = "*"
100 A$(4) = ":"
110 T = 0                   :REM BEURTTELLER
120 N = 1                   :REM NIVEAU
130 P = 0                   :REM NIVEAUINDIKATOR
140 Y = 0                   :REM PERC.GOEDEN
150 G = 0                   :REM GOEDE ANTWOORDEN
160 RETURN
170 :
180 REM *** UITVOER ***
190 CLS                     :REM SCHERM WISSEN
200 PRINT "*** SOMMEN ***"
210 PRINT "BEURT            :";T
220 PRINT "NIVEAU           :";N
230 PRINT "PERC. GOEDEN     :";Y
240 PRINT
250 ON K GOSUB 300  ,340
260 PRINT
270 RETURN
280 :
```

```
290 REM *** PRINT REGEL K = 1 ***
300 PRINT A; A$(J);B;"= .."
310 RETURN
320 :
330 REM *** PRINT REGEL K = 2 ***
340 PRINT A;A$(J);" .. = ";B
350 RETURN
360 :
370 REM *** GETALLEN TREKKEN ***
380 A = INT(RND(0) * X + 1)
390 B = INT(RND(0) * X + 1)
400 RETURN
410 :
420 REM *** VRAAG OPHALEN ***
430 N = INT(P/5 + 1)
440 IF N < 9 THEN 460
450 N = 8
460 J = N
470 IF J < 5 THEN 490
480 J = 4
490 J = INT(RND(0) * J + 1)
500 ON J GOSUB 590  ,650  ,730  ,820
510 K = INT(RND(0) * 2 +1)
520 IF K = 1 THEN 560
530 H = B
540 B = C
550 C = H
560 RETURN
570 :
580 REM *** OPTELLEN ***
590 X = INT(N/2 + 1) * 10
600 GOSUB 380
610 C = A + B
620 RETURN
630 :
640 REM *** AFTREKKEN ***
650 X = INT(N/2 + 1) * 10
660 GOSUB 380
670 C = A + B
680 A = C
690 C = A - B
700 RETURN
710 :
720 REM *** VERMENIGVULDIGEN ***
730 X = 10
740 IF N < 5 THEN 760
750 X = X + N
760 GOSUB 380
770 IF A > 10 OR B > 10 THEN 760
780 C = A * B
790 RETURN
800 :
810 REM *** DELEN ***
820 X = 10
830 IF N < 5 THEN 850
```

```
840 X = X + N
850 GOSUB 380
860 IF A > 10 OR B > 10 THEN 850
870 C = A * B
880 A = C
890 C = A/B
900 RETURN
910 :
920 REM *** INVOER ***
930 PRINT "STOP = -9 !"
940 PRINT "WAT MOET ER INGEVULD WORDEN ";
950 INPUT Q
960 IF Q = -9 THEN 1140
970 T = T + 1
980 P = P -1
990 H = 0
1000 A$ = "=> FOUT <="
1010 IF Q <> C THEN 1060
1020 A$ = "=> GOED <="
1030 H = 1
1040 G = G + 1
1050 P = P + 2
1060 IF P > 0 THEN 1080
1070 P = 1
1080 Y = INT((G/T) * 1000)/10
1090 PRINT TAB(10) A$
1100 PRINT "HET ANTWOORD WAS ";C
1110 REM ** WACHTLUS **
1120 FOR I = 1 TO 250 * (2 - H)
1130 NEXT I
1140 RETURN
1150 :
1160 REM *** RESULTAAT ***
1170 CLS                      :REM SCHERM WISSEN
1180 PRINT "*** SOMMEN ***
1190 PRINT
1200 PRINT "TOTAAL AANTAL BEURTEN  : ";T
1210 PRINT "TOTAAL AANTAL GOEDEN   : ";G
1220 PRINT "PERC. GOEDE ANTWOORDEN : ";Y
1230 PRINT "NIVEAU                 : ";N
1240 RETURN
1250 :
1260 REM **** STUURMODULE ****
1270 REM *** INITIALISEREN ***
1280 GOSUB 70
1290 REM *** VRAAG OPHALEN ***
1300 GOSUB 430
1310 REM *** UITVOER ***
1320 GOSUB 190
1330 REM *** INVOER ***
1340 GOSUB 930
1350 IF Q <> -9 THEN 1300
1360 REM *** RESULTAAT ***
1370 GOSUB 1170
1380 END
```

Voorbeeld

```
*** SOMMEN ***
BEURT           : 5
NIVEAU          : 1
PERC. GOEDEN    : 80

 9 + 4 = ..

STOP = -9 !
WAT MOET ER INGEVULD WORDEN  13
          => GOED <=
HET ANTWOORD WAS  13


*** SOMMEN ***
BEURT           : 6
NIVEAU          : 2
PERC. GOEDEN    : 83.3

 34 - .. =  20

STOP = -9 !
WAT MOET ER INGEVULD WORDEN  14
          => GOED <=
HET ANTWOORD WAS  14


*** SOMMEN ***
BEURT           : 7
NIVEAU          : 2
PERC. GOEDEN    : 85.7

 19 - 11 = ..

STOP = -9 !
WAT MOET ER INGEVULD WORDEN -9


*** SOMMEN ***
TOTAAL AANTAL BEURTEN   :  7
TOTAAL AANTAL GOEDEN    :  6
PERC. GOEDE ANTWOORDEN :  85.7
NIVEAU                  :  2
```

2.10 Wat is het volgende getal?

Dit is een heel verraderlijk programma dat niet alleen voor kinderen, maar ook voor volwassenen geschikt is. Op de lage niveaus is het nog vrij gemakkelijk. Men kan dan nog wel in één oogopslag zien wat het volgende getal moet zijn. Komt het programma op hogere niveaus dan kunnen de getallen uit twee verschillende rijen bestaan en dan wordt het erg moeilijk te bepalen wat het volgende getal zal zijn. Er zijn maar weinig mensen geweest die het hoogste niveau in het programma hebben bereikt. Er zijn er nog minder die zich op het hoogste niveau hebben kunnen handhaven.
De werking van het programma is als volgt. Er worden zes getallen achter elkaar op het beeldscherm gezet. In plaats van het zevende getal wordt er een vraagteken neergezet. De bedoeling is dit zevende getal af te leiden uit de vorige zes getallen of minder. Geeft men steeds het goede antwoord dan zal het niveau langzaam stijgen en worden de vragen ingewikkelder. Maakt men daarentegen fouten, dan zakt langzaam het niveau. De soort reeks, die wordt opgebouwd, is volkomen willekeurig, zodat het programma blijft boeien hoe vaak men het ook speelt.
Het programma toont altijd de uitkomst; mocht dit te kort of te lang zijn, dan kan men de lengte van de wachtlus in regel 960 aanpassen.
De kleine formule tussen haakjes zorgt ervoor dat bij een fout antwoord de wachttijd twee keer zo lang is als bij een goed antwoord.

Programma

```
10 REM *** WAT IS HET VOLGENDE GETAL ***
20 RANDOM                  :REM DIT STATEMENT MAG VERVALLEN
30 DIM A(7)
40 GOTO 1270
50 :
60 REM *** INITIALISEREN ***
70 T = 0                   :REM AANTAL BEURTEN
80 G = 0                   :REM AANTAL GOEDE ANTW.
90 Y = 0                   :REM PERC. GOEDEN
100 P = 0                  :REM NIVEAU-INDICATOR
110 N = 1                  :REM NIVEAU
120 RETURN
130 :
140 REM *** OPLOPEND NIVEAU <= 1 ***
150 FOR I = 1 TO 7
160 A(I) = B + R * I
170 NEXT I
180 RETURN
190 :
200 REM *** AFDALEND NIVEAU <= 2 ***
210 GOSUB 150             :REM OPLOPEND
220 FOR I = 1 TO 3
230 A = A(8-I)
240 A(8-I) = A(I)
250 A(I) = A
260 NEXT I
270 RETURN
280 :
290 REM *** TWEE REEKSEN ***
```

```
300 REM NIVEAU <= 3 EN NIVEAU <= 4
310 FOR J = 1 TO 2
320 B = INT(RND(0) * X + 1)
330 R = INT(RND(0) * Z + 1)
340 IF RND(0) > 0.5 OR N = 3 THEN 360
350 R = R * -1
360 GOSUB 410
370 NEXT J
380 RETURN
390 :
400 REM *** OPLOPEND MET STAPGROOTTE 2 ***
410 S = B
420 FOR I = J TO 8 -J STEP 2
430 A(I) = S + R
440 S = A(I)
450 NEXT I
460 RETURN
470 :
480 REM *** RANDOM GEVULD 1 NIVEAU <= 5 ***
490 FOR I = 1 TO 6
500 A(I) = INT(RND(0) * X + 1)
510 NEXT I
520 J = 1
530 GOSUB 410
540 RETURN
550 :
560 REM *** RANDOM GEVULD NIVEAU <= 6 ***
570 H = INT(RND(0) * 2 + 1)     :REM OPLOPEND/AFDALEND ?
580 ON H GOSUB 150  ,210
590 Q = INT(RND(0) * 3 + 1)
600 FOR I = 1 TO Q
610 H = INT(RND(0) * 6 + 1)
620 A(H) = INT(RND(0) * X)
630 NEXT I
640 RETURN
650 :
660 REM *** UITVOER ***
670 CLS                       :REM SCHERM WISSEN
680 PRINT "*** WAT IS HET VOLGENDE GETAL ***"
690 PRINT
700 PRINT "NIVEAU          :";N
710 PRINT "AANTAL BEURTEN :";T+1
720 PRINT "PERC. GOEDEN    :";Y
730 PRINT
740 FOR I = 1 TO 6
750 PRINT A(I);
760 NEXT I
770 PRINT " .?."
780 RETURN
790 :
800 REM *** INVOER / VERWERKING ***
810 PRINT "WAT IS HET VOLGENDE GETAL <STOP = 9999> ";
820 INPUT E
830 IF E = 9999 THEN 1050
840 H = 0                     :REM STUURVARIABELE
```

```
850 A$ = "=> FOUT <="
860 P = P - 1             :REM NIVEAU-INDICATOR
870 T = T + 1             :REM BEURT + 1
880 IF E <> A(7) THEN 930
890 P = P + 2
900 H = 1
910 A$ = "=> GOED <="
920 G = G + H
930 PRINT A$
940 PRINT "HET GETAL WAS ";A(7)
950 REM ** WACHTLUS **
960 FOR I = 1 TO 250 * (2 - H)
970 NEXT I
980 REM ** RESULTAAT VERWERKING **
990 N = INT(P/5)+1
1000 IF N < 7 THEN 1020
1010 N = 6
1020 IF N > 0 THEN 1040
1030 N = 1
1040 Y = INT((G/T) * 1000) / 10   :REM PERC. GOED
1050 RETURN
1060 :
1070 REM *** VRAAG OPHALEN ***
1080 X = N * 10
1090 Z = N * 5
1100 B = INT(RND(0) * X + 1)
1110 R = INT(RND(0) * Z + 1)
1120 H = INT(RND(0) * N + 1)
1130 ON H GOSUB 150  ,210  ,310  ,310  ,490  ,570
1140 RETURN
1150 :
1160 REM *** RESULTAAT ***
1170 CLS
1180 PRINT "*** WAT IS HET VOLGENDE GETAL ***"
1190 PRINT
1200 PRINT "AANTAL BEURTEN          :";T
1210 PRINT "AANTAL GOEDE ANTWOORDEN :";G
1220 PRINT "PERCENTAGE GOED         :";Y
1230 PRINT"NIVEAU                   :";N
1240 RETURN
1250 :
1260 REM *** INITIALISEREN ***
1270 GOSUB 70
1280 REM *** VRAAG OPHALEN ***
1290 GOSUB 1080
1300 REM *** UITVOER ***
1310 GOSUB 670
1320 REM *** INVOER / VERWERKING ***
1330 GOSUB 810
1340 IF E <> 9999 THEN 1290
1350 REM *** RESULTAAT ***
1360 GOSUB 1170
1370 END
```

Voorbeeld

```
*** WAT IS HET VOLGENDE GETAL ***

NIVEAU          : 1
AANTAL BEURTEN  : 1
PERC. GOEDEN    : 0

 6  8  10  12  14  16  .?.
WAT IS HET VOLGENDE GETAL <STOP = 9999>  17
=> FOUT <=
HET GETAL WAS  18


*** WAT IS HET VOLGENDE GETAL ***

NIVEAU          : 1
AANTAL BEURTEN  : 2
PERC. GOEDEN    : 0

 10  15  20  25  30  35  .?.
WAT IS HET VOLGENDE GETAL <STOP = 9999>  40
=> GOED <=
HET GETAL WAS  40


*** WAT IS HET VOLGENDE GETAL ***

NIVEAU          : 1
AANTAL BEURTEN  : 5
PERC. GOEDEN    : 75

 4  6  8  10  12  14  .?.
WAT IS HET VOLGENDE GETAL <STOP = 9999>  9999


*** WAT IS HET VOLGENDE GETAL ***

AANTAL BEURTEN            : 4
AANTAL GOEDE ANTWOORDEN   : 3
PERCENTAGE GOED           : 75
NIVEAU                     : 1
```

3. Spelprogramma's

3.1 Struisvogelrace

Eens per jaar worden in Boomcity de grote struisvogelraces gehouden. Dit zijn de grootste races en alleen de hoogste in het klassement van de regionale competitie worden geplaatst. Illustere namen als Whooper, Langzame Jenny en Driftige Floeper zullen aan de start verschijnen.
Zoals gebruikelijk is, wordt er bij dit soort races behoorlijk gegokt en iedereen doet mee. De te winnen prijzen liegen er niet om.
Als buitenstaander lijkt het verstandig bij de eerste races voorzichtig in te zetten. Een van de vreemde regels bij de struisvogelraces is dat men van het terrein verwijderd wordt als men alles verloren heeft.



Het programma maakt gebruik van een randomgenerator om de race te simuleren. Omdat het programma in standaard BASIC geschreven is, wordt iedere keer het beeldscherm gewist en opnieuw opgebouwd. Dit programma leent zich uitstekend om de nieuwe spelsituatie in de renbaan te POKEn. Men moet dan niet vergeten de vorige plaatsen te wissen.
Voor iemand, die net met microcomputers begint, geeft dit programma een aardig beeld hoe men met strings kan manipuleren.

Programma

```
10 REM *** STRUISVOGELRACE ***
20 RANDOM                         :REM DIT STATEMENT KAN VERVALLEN
30 DIM A(6,2), B$(6)
40 DATA WEGRACER,6,WOOPER,3,"LANGZAME JENNY",10
50 DATA "DRIFTIGE FLOEPER",5,"STRUISE BOL", 12
60 DATA "LOPEND POTLOOD",4
70 GOTO 1110
80 :
90 REM *** INITIALISEREN ***
100 R = 0                   :REM AANTAL RACES
110 B = 1000                :REM TOTAAL AAN INZET
120 REM ** INTERNE DATA-FILE LEZEN **
130 RESTORE
140 FOR I = 1 TO 6
150 READ B$(I),A(I,2)
160 NEXT I
170 RETURN
180 :
190 REM *** INVOER ***
200 CLS                    :REM SCHERM WISSEN
210 PRINT "*** STRUISVOGELRACE ***"
220 PRINT "RACE-NUMMER ";R+1 ;TAB(20)"MAXIMALE INZET ";B
230 PRINT
240 FOR I = 1 TO 6
250 PRINT I;B$(I);TAB(25)A(I,2);TAB(29)": 1"
260 NEXT I
270 PRINT
280 PRINT "WAT IS UW KEUZE <STOP = 0> ";
290 INPUT K
300 IF K = 0 THEN 420
310 IF K < 0 OR K > 6 THEN 200
320 PRINT "HOEVEEL ZET U IN OP ";B$(K)
330 PRINT "U KUNT MAXIMAAL ";B;"INZETTEN ! HOEVEEL ";
340 INPUT L
350 IF L < 0 OR L > B THEN 320
360 B = B - L
370 REM ** START RACE **
380 FOR I = 1 TO 6
390 A(I,1) = 1                :REM STRUISVOGEL OP START
400 NEXT I
410 R = R + 1
420 RETURN
430 :
440 REM *** BEWERKING ***
450 X = 1
460 FOR I = 1 TO 6
470 A(I,1) = A(I,1) + RND(0) * 3
480 IF A(X,1) > A(I,1) THEN 500
490 X = I                    :REM WISSELEN
500 NEXT I
510 REM ** DE FINISH GEHAALD ? **
520 H = 0
530 IF A(X,1) < 32 THEN 550
540 H = 1                    :REM RACE GEEINDIGD
```

```
550 RETURN
560 :
570 REM *** UITVOER ***
580 CLS                       :REM SCHERM WISSEN
590 PRINT "*** STRUISVOGELRACE ***"
600 PRINT "RACE-NUMMER ";R;TAB(17)"INGEZET :";L;
610 PRINT "TOTAAL ";B
620 PRINT "INGEZET OP STRUISVOGELNR.";K;B$(K)
630 PRINT
640 PRINT "  !--!--!--!--!--!--!--!--!--!--!"
650 FOR I = 1 TO 6
660 D$ = LEFT$("FINISH",I)
670 D$ = RIGHT$(D$,1)
680 E$ = LEFT$("123456",I)
690 E$ = RIGHT$(E$,1)
700 Y = INT(A(I,1) + 0.5)
710 IF Y > 32 THEN 740
720 PRINT TAB(Y)E$;       :REM STRUISVOGELNR.
730 IF Y = 32 THEN 770
740 PRINT TAB(32)D$;
750 IF Y < 32 THEN 770
760 PRINT TAB(Y)E$;       :REM STRUISVOGELNR.
770 D$ = "<-"
780 IF X = I THEN 800
790 D$ = ""
800 PRINT TAB(35)D$
810 NEXT I
820 PRINT "  !--!--!--!--!--!--!--!--!--!--!"
830 RETURN
840 :
850 REM *** RESULTAAT ***
860 REM ** VERLOREN ? **
870 Z = L * -1
880 B$ = "VERLOREN : "
890 IF X <> K THEN 940
900 Z = L * A(K,2)             :REM UITKERING
910 B$ = "GEWONNEN : "
920 B = B + Z
930 REM ** UITVOER **
940 GOSUB 580
950 PRINT "KOERSWINNAAR IS NUMMER";X;B$(X)
960 PRINT B$;ABS(Z)
970 PRINT "NOG EEN KOERS <JA/NEE> ";
980 INPUT C$
990 C$ = LEFT$(C$,1)
1000 H = 0
1010 IF C$ = "N" THEN 1040
1020 H = 1
1030 IF C$ <> "J" THEN 970
1040 IF B > 0 THEN 1070
1050 PRINT "ALLES VERLOREN !"
1060 H = 0
1070 RETURN
1080 :
1090 REM **** STUURMODULE ***
```

```
1100 REM *** INITIALISEREN ***
1110 GOSUB 100
1120 REM *** INVOER ***
1130 GOSUB 200
1140 IF K = 0 THEN 1230
1150 REM *** UITVOER ***
1160 GOSUB 580
1170 REM *** BEWERKING ***
1180 GOSUB 450
1190 IF H = 0 THEN 1160
1200 REM *** RESULTAAT ***
1210 GOSUB 870
1220 IF H = 1 THEN 1130
1230 END
```

Voorbeeld

```
*** STRUISVOGELRACE ***
RACE-NUMMER  1      MAXIMALE INZET  1000

 1 WEGRACER               6  : 1
 2 WOOPER                 3  : 1
 3 LANGZAME JENNY         10 : 1
 4 DRIFTIGE FLOEPER       5  : 1
 5 STRUISE BOL            12 : 1
 6 LOPEND POTLOOD         4  : 1

WAT IS UW KEUZE <STOP = 0>  5
HOEVEEL ZET U IN OP STRUISE BOL
U KUNT MAXIMAAL  1000 INZETTEN ! HOEVEEL  500
*** STRUISVOGELRACE ***
RACE-NUMMER  1   INGEZET : 500 TOTAAL  500
INGEZET OP STRUISVOGELNR. 5 STRUISE BOL

  !--!--!--!--!--!--!--!--!--!--!
 1                              F
 2                              I
 3                              N
 4                              I
 5                              S
 6                              H
  !--!--!--!--!--!--!--!--!--!--!
*** STRUISVOGELRACE ***
RACE-NUMMER  1   INGEZET : 500 TOTAAL  500
INGEZET OP STRUISVOGELNR. 5 STRUISE BOL

  !--!--!--!--!--!--!--!--!--!--!
                     1          F  <-
                   2            I
                3               N
                 4              I
                    5           S
              6                 H
  !--!--!--!--!--!--!--!--!--!--!
*** STRUISVOGELRACE ***
RACE-NUMMER  1   INGEZET : 500 TOTAAL  6500
INGEZET OP STRUISVOGELNR. 5 STRUISE BOL

  !--!--!--!--!--!--!--!--!--!--!
                                1
                               2I
                              3 N
                            4   I
                                5  <-
                         6      H
  !--!--!--!--!--!--!--!--!--!--!
KOERSWINNAAR IS NUMMER 5 STRUISE BOL
GEWONNEN :   6000
NOG EEN KOERS <JA/NEE> NEE
```

3.2 Fruitmachine

Overal ziet men ze staan; de fruitmachines. In het café, de snackbar, bij de kapper, de Chinees en nu thuis op de computer. Een verschil is wel; in deze fruitmachine hoeft geen geld gegooid te worden, maar hij keert ook niets uit! De computer laat u ook niet vrij om grof te gokken want de inzet kan niet kleiner dan 0 of groter dan 5 zijn.
De puntentelling geschiedt aan de hand van onderstaand tabelletje.

JOKER	JOKER	JOKER	1:20
BAR	BAR	BAR	1:15
KLOK	KLOK	KLOK	1:12
CITROEN	CITROEN	CITROEN	1: 8
APPEL	APPEL	APPEL	1: 5
KERS	KERS	–	1: 3
KERS	–	–	1: 2

Het programma werkt niet met plaatjes, maar met de benamingen van de plaatjes. Het is natuurlijk veel leuker om met plaatjes te werken. Op sommige computers is dit vrij eenvoudig te realiseren. Maar men zal altijd PEEK- en POKE-opdrachten nodig hebben om tot de gewenste resultaten te komen. Verder zou het aardig zijn te suggereren dat de plaatjes door het beeldscherm draaien. Niet bepaald een eenvoudige opgaaf om dit voor elkaar te krijgen.

Programma

```
10 REM *** FRUITMACHINE ***
20 DIM A$(6), B(3), D(10,2)
30 RANDOM
40 DATA "-  JOKER  -","-   BAR   -","-  KLOK   -"
50 DATA "- CITROEN -","-  APPEL  -","-  KERS   -"
60 DATA 111,20, 221,15, 333,12, 331,12, 444,8
70 DATA 441,8, 555,5, 551,5 ,66,3, 6,2
80 GOTO 760
90 :
100 REM *** INITIALISEREN ***
110 RESTORE
120 REM ** SYMBOLEN LEZEN **
130 FOR I = 1 TO 6
140 READ A$(I)
150 NEXT I
160 REM ** WAARDERING LEZEN **
170 FOR I = 1 TO 10
180 READ D(I,1), D(I,2)
190 NEXT I
200 T = 0
210 B = 0
220 RETURN
230 :
240 REM *** DRAAIEN ***
250 Y = INT(RND(0) * 10 + 1)
260 FOR I = 1 TO Y
270 S = 0
280 FOR J = 1 TO 3
290 B(J) = INT(RND(0) * 6 + 1)
300 S = S * 10 + B(J)
310 NEXT J
320 NEXT I
330 RETURN
340 :
350 REM *** RESULTAAT ***
360 H = 0                   :REM STUURVARIABELE
370 B = B * -1
380 IF B = 0 THEN 510
390 FOR I = 1 TO 8
400 IF S <> D(I,1) THEN 430
410 H = I
420 I = 8                   :REM LUSTELLER MAX.
430 NEXT I
440 IF H <> 0 THEN 490
450 H = 9
460 IF D(H,1) = B(1) * 10 + B(2) THEN 490
470 H = 10
480 IF D(H,1) <> B(1) THEN 500
490 B = D(H,2) * ABS(B)
500 T = T + B               :REM TOTAAL WINST
510 RETURN
520 :
530 REM *** UITVOER ***
```

```
540 CLS                          :REM SCHERM WISSEN
550 PRINT "*** FRUITMACHINE ***"
560 PRINT
570 PRINT "DEZE BEURT :"B
580 PRINT "TOTAAL     :"T
590 PRINT "   !----------!!----------!!----------!"
600 FOR I = 1 TO 3
610 PRINT TAB(I* 11 - 8)A$(B(I));
620 NEXT I
630 PRINT
640 PRINT "   !----------!!----------!!----------!"
650 PRINT
660 RETURN
670 :
680 REM *** INVOER ***
690 PRINT "HOEVEEL ZET JE IN  <STOP = 0> ";
700 INPUT B
710 IF B < 0 OR B > 5 THEN 690
720 RETURN
730 :
740 REM **** STUURMODULE ****
750 REM *** INITIALISEREN ***
760 GOSUB 110
770 REM *** DRAAIEN ***
780 GOSUB 250
790 REM *** RESULTAAT ***
800 GOSUB 360
810 REM *** UITVOER ***
820 GOSUB 540
830 REM *** INVOER ***
840 GOSUB 690
850 IF B <> 0 THEN 780
860 END
```

Voorbeeld

```
*** FRUITMACHINE ***

DEZE BEURT : 0
TOTAAL     : 0
    !----------!!----------!!----------!
    -  JOKER   --  APPEL   --  JOKER   -
    !----------!!----------!!----------!

HOEVEEL ZET JE IN  <STOP = 0>  25
HOEVEEL ZET JE IN  <STOP = 0>  2


*** FRUITMACHINE ***

DEZE BEURT :-2
TOTAAL     :-2
    !----------!!----------!!----------!
    -  JOKER   --  APPEL   -- CITROEN -
    !----------!!----------!!----------!

HOEVEEL ZET JE IN  <STOP = 0>  2


*** FRUITMACHINE ***

DEZE BEURT :-2
TOTAAL     :-4
    !----------!!----------!!----------!
    -  JOKER   --   BAR    --  KLOK    -
    !----------!!----------!!----------!

HOEVEEL ZET JE IN  <STOP = 0>  2


*** FRUITMACHINE ***

DEZE BEURT : 4
TOTAAL     : 0
    !----------!!----------!!----------!
    -  KERS    --  JOKER   --  APPEL   -
    !----------!!----------!!----------!

HOEVEEL ZET JE IN  <STOP = 0>  0
```

3.3 Torens van Hanoi

Het probleem 'Torens van Hanoi' is al een vrij oud probleem. Veel mensen hebben het trachten op te lossen. Het lijkt zo simpel, alleen maar een stel schijven overhevelen. In de praktijk blijkt het zeker niet mee te vallen. Men moet zich goed concentreren om geen fouten te maken. Gebruikt men drie schijven dan zal men een keer of acht moeten schuiven met de schijven. Neemt men zeven schijven dan moet men rekenen op minstens 128 keer schuiven. Het is maar een waarschuwing dat het niet eenvoudig is.
Met moet proberen de schijven van staaf 1 naar staaf 3 te krijgen, waarbij men staaf 2 als hulpstaaf mag gebruiken. Er geldt een regel: Een grotere schijf mag nooit bovenop een kleinere schijf komen te liggen.
Wanneer men het programma start, kan men kiezen met hoeveel schijven men wil spelen. Minimaal zijn dit er drie, maximaal zijn dit er zeven. Men geeft dan op van welke staaf men een schijf wil plaatsen en dan op welke staaf men hem plaatsen wil. Geeft men bij de eerste vraag een nul, dan neemt het programma aan dat men klaar is. Het programma gaat na of op staaf 3 de schijven in de goede volgorde liggen. Zondigt men onderweg tegen de regels van het spel, dan stopt het programma direct en zal men opnieuw moeten beginnen.



In dit programma moet de speler de oplossing van het probleem zien te vinden, maakt hij een fout dan kan de computer niet vóórdoen hoe het moet. Het zou best aardig zijn als de

computer dit wel kon. Dit is te realiseren door een aantal modules aan het programma toe te voegen. In BASIC zal dit een behoorlijk probleem blijken te zijn. In het boek 'Structured computer organization' van Tanenbaum, een uitgave van Prentice Hall, staat een idee hoe men de oplossing kan vinden. Het blijkt dat men een stackmechanisme nodig heeft. Een stack kenmerkt zich door het LIFO-principe. Last In First Out.

Programma

```
10 REM *** TORENS VAN HANOI ***
20 DIM A(8,3)
30 GOTO 1060
40 :
50 REM *** INITIALISEREN ***
60 S = 15
70 FOR I = 1 TO 8
80 A(I,1) = S - 2 * I
90 A(I,2) = 0
100 A(I,3) = 0
110 NEXT I
120 B = 1
130 Q = 7
140 A$ = "*************"
150 RETURN
160 :
170 REM *** INVOER I ***
180 GOSUB 360              :REM *** UITVOER ***
190 PRINT "HOEVEEL SCHIJVEN <MIN.=3 MAX.=7> ";
200 INPUT Q
210 IF Q < 3 OR Q > 7 THEN 180
220 IF Q = 7 THEN 320
230 REM ** GEDEELTELIJK OPNIEUW INITIALISEREN ***
240 S = A(7 - Q,1)
250 FOR I = Q TO 7
260 A(I,1) = 0             :REM STAAF SCHOONMAKEN
270 NEXT I
280 REM * SCHIJVEN PLAATSEN *
290 FOR I = 1 TO Q
300 A(I,1) = S - 2 * I
310 NEXT I
320 A(8,1) = Q
330 RETURN
340 :
350 REM *** UITVOER ***
360 CLS
370 PRINT "*** TORENS VAN HANOI ***"
380 PRINT "BEURT ";B
390 PRINT
400 PRINT TAB(5)1; TAB(19)2; TAB(33)3
410 PRINT
420 FOR I = 7 TO 1 STEP -1
430 FOR J = 1 TO 3
440 B$ = "I"
450 V = 6
460 IF A(I,J) = 0 THEN 490
470 V = (13 - A(I,J))/2
480 B$ = RIGHT$(A$,A(I,J))
490 P = J * 14 - 14 + V
500 PRINT TAB(P)B$;
510 NEXT J
520 PRINT
530 NEXT I
```

```
540 RETURN
550 :
560 REM *** INVOER II ***
570 PRINT "VAN STAAFNUMMER <STOP = 0> ";
580 INPUT D
590 IF D = 0 THEN 640
600 IF D < 0 OR D > 3 THEN 570
610 PRINT "NAAR STAAFNUMMER ";
620 INPUT F
630 IF F < 0 OR F > 3 THEN 610
640 RETURN
650 :
660 REM *** BEWERKING ***
670 H = 1
680 B = B + 1
690 IF A(8,D) = 0 OR A(8,F) = Q THEN 800
700 H = 0
710 REM ** SCHIJF VAN STAAF-D HALEN **
720 C = A(8,D)
730 A(8,D) = A(8,D) - 1
740 P = A(C,D)
750 A(C,D) = 0
760 REM ** SCHIJF OP STAAF-F ZETTEN **
770 A(8,F) = A(8,F) + 1
780 C = A(8,F)
790 A(C,F) = P
800 RETURN
810 :
820 REM *** CONTROLE ***
830 IF A(8,F) = 1 AND D <> 0 THEN 940
840 IF D <> 0 THEN 860
850 F = 3
860 FOR I = 1 TO A(8,F) - 1
870 IF A(I,F) > A(I + 1,F) THEN 900
880 H = 1
890 I = A(8,F)
900 NEXT I
910 IF H = 1 OR D <> 0 THEN 940
920 IF A(8,F) = Q THEN 940
930 H = 1
940 RETURN
950 :
960 REM *** RESULTAAT ***
970 GOSUB 360              :REM *** UITVOER ***
980 B$ = "FOUT ! <="
990 IF H = 1 THEN 1010
1000 B$ = "GOED ! <="
1010 PRINT "HET IS ";B$
1020 RETURN
1030 :
1040 REM **** STUURMODULE ****
1050 REM *** INITITIALISEREN ***
1060 GOSUB 60
1070 REM *** INVOER I **
1080 GOSUB 180
```

```
1090 REM *** UITVOER ***
1100 GOSUB 360
1110 REM *** INVOER ***
1120 GOSUB 570
1130 IF D = 0 THEN 1180
1140 REM *** BEWERKING ***
1150 GOSUB 670
1160 IF H = 1 THEN 1210
1170 REM *** CONTROLE ***
1180 GOSUB 830
1190 IF D <> 0 AND H <> 1 THEN 1100
1200 REM *** RESULTAAT ***
1210 GOSUB 970
1220 END
```

Voorbeeld

```
*** TORENS VAN HANOI ***
BEURT  1

      1             2             3

      *             I             I
     ***            I             I
    *****           I             I
   *******          I             I
  *********         I             I
 ***********        I             I
*************       I             I
HOEVEEL SCHIJVEN <MIN.=3 MAX.=7>  3

*** TORENS VAN HANOI ***
BEURT  1

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      *             I             I
     ***            I             I
    *****           I             I
VAN STAAFNUMMER <STOP = 0>  1
NAAR STAAFNUMMER  3

*** TORENS VAN HANOI ***
BEURT  2

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
     ***            I             I
    *****           I             *
VAN STAAFNUMMER <STOP = 0>  1
NAAR STAAFNUMMER  2

*** TORENS VAN HANOI ***
BEURT  3

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
    *****          ***            *
VAN STAAFNUMMER <STOP = 0>  3
NAAR STAAFNUMMER  2

*** TORENS VAN HANOI ***
BEURT  4

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             *             I
    *****          ***            I
VAN STAAFNUMMER <STOP = 0>  1
NAAR STAAFNUMMER  3

*** TORENS VAN HANOI ***
BEURT  5

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             *             I
      I            ***          *****
VAN STAAFNUMMER <STOP = 0>  2
NAAR STAAFNUMMER  1

*** TORENS VAN HANOI ***
BEURT  6

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      *            ***          *****
VAN STAAFNUMMER <STOP = 0>  2
NAAR STAAFNUMMER  3

*** TORENS VAN HANOI ***
BEURT  7

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I            ***
      *             I           *****
VAN STAAFNUMMER <STOP = 0>  1
NAAR STAAFNUMMER  3
```

```
*** TORENS VAN HANOI ***
BEURT  7

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             *
      I             I            ***
      I             I           *****
VAN STAAFNUMMER <STOP = Ø>  Ø

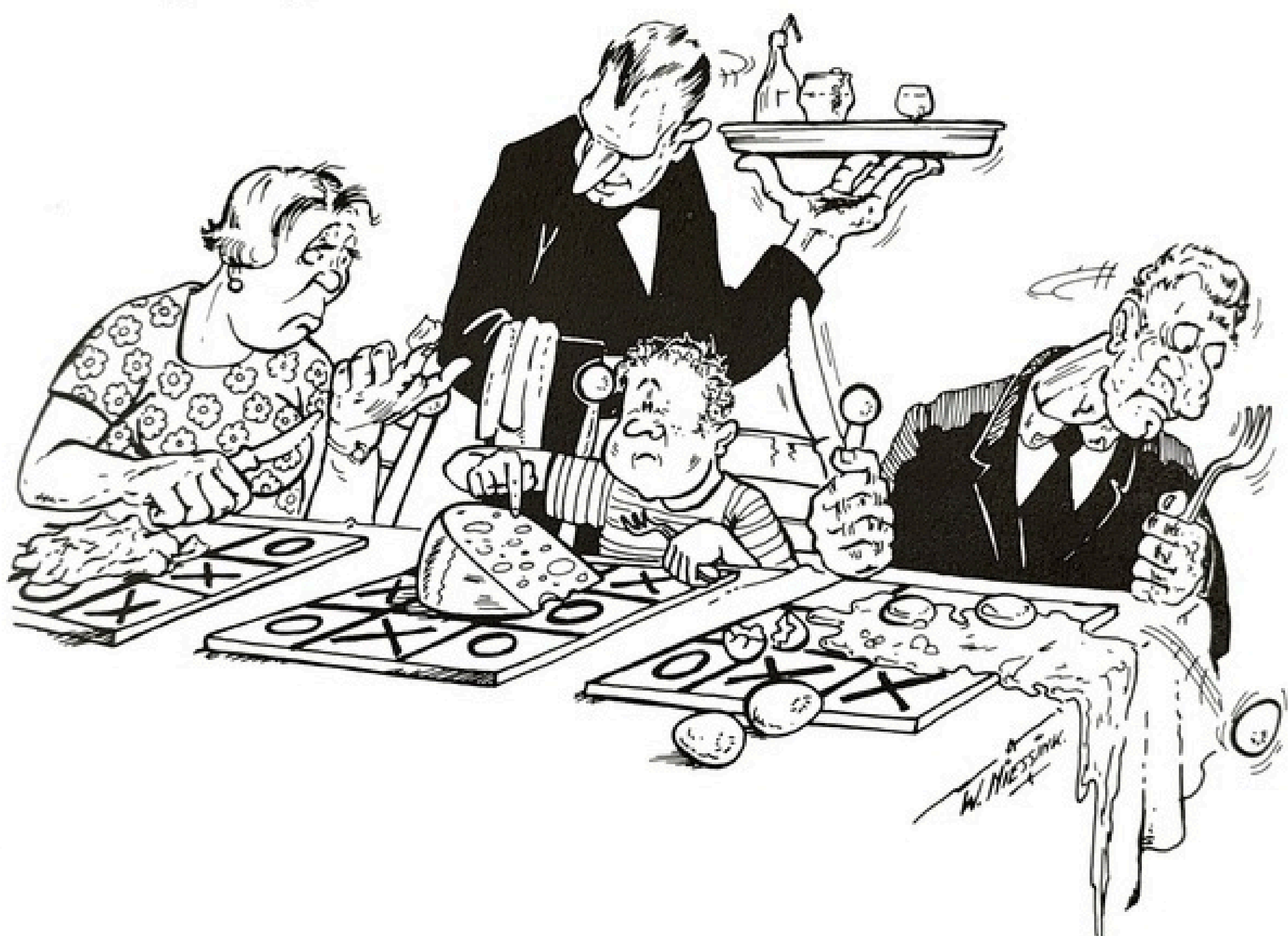
*** TORENS VAN HANOI ***
BEURT  7

      1             2             3

      I             I             I
      I             I             I
      I             I             I
      I             I             I
      I             I             *
      I             I            ***
      I             I           *****
HET IS GOED ! <=
```

3.4 Boter, kaas en eieren

Boter, kaas en eieren is een vrij eenvoudig spelletje, waarbij men goed moet opletten geen fouten te maken. Maakt men een fout dan heeft de tegenstander gewonnen. Een tegenstander die geen fouten maakt, is de computer. Voor veel mensen is het onbegrijpelijk dat een computer zo'n spel kan spelen. De computer neemt beslissingen en doet goede zetten. Voor sommige mensen is dit zelfs beangstigend.
Ondanks het vrij eenvoudige spel, is het programma toch nog vrij ingewikkeld. Er moet namelijk voor worden gezorgd dat het programma alle spelsituaties aankan en op grond daarvan ook tot een goede beslissing komt. Gelukkig zijn de mogelijke zetten beperkt.
De speler of de computer begint als eerste. De keuze wordt aan de speler overgelaten.
Aan de hand van de programmalisting is gemakkelijk na te gaan, hoe de machine tot een beslissing komt. Het blijkt dat het programma alle mogelijke situaties afloopt en op grond van de resultaten een zet doet. Dit programma is dan ook een bescheiden voorbeeld van kunstmatige intelligentie.



Na iedere zet wordt het beeldscherm gewist en daarna opnieuw opgebouwd met de gewijzigde spelsituatie. Het nadeel is dat er een zeer onrustig beeld ontstaat. Met behulp van POKE-opdrachten kan men de wijzigingen rechtstreeks op het beeldscherm veranderen. Beschikt men over een kleurencomputer dan kan men met kleuren het speelveld verlevendigen. Verder verdient het zeker aanbeveling het gehele speelveld te vergroten. Dit heeft echter wel de nodige consequenties voor de uitvoermodule.

Programma

```
10 REM *** BOTER KAAS EN EIEREN ***
20 DATA 1,2,3,4,5,6,7,8,9,1,4,7,2,4,8
30 DATA 3,6,9,1,5,9,3,5,7
40 DIM A(9),B(3)
50 GOTO 1350
60 :
70 REM *** INITIALISEREN ***
80 FOR I = 1 TO 9
90 A(I) = 0
100 NEXT I
110 B = 0 : C = 0 : F = 0 : G = 10
120 A$ = "!---!---!---!"
130 :
140 RETURN
150 :
160 REM *** TEGENZET COMPUTER ***
170 REM ** BEURT 1 COMPUTER **
180 P = 0 : C = C + 1 : G = G - 1
190 IF C > 1 THEN 290
200 IF A(7) = 9 THEN P = 3
210 IF A(9) = 9 THEN P = 1
220 IF A(3) = 9 THEN P = 1
230 IF A(1) = 9 THEN P = 9
240 IF P <> 0 THEN 540
250 P = 5
260 IF A(P) = 0 THEN 290
270 P = 9
280 REM ** BEURT 2 COMPUTER **
290 IF B > 1 OR P <> 0 THEN 330
300 P = 1
310 IF A(1) = 0 AND A(9) = 0 THEN 540
320 P = 3
330 IF P <> 0 THEN 540
340 REM ** COMPUTER WINT ? **
350 R = 1 : Y = 1
360 GOSUB 930
370 IF P <> 0 THEN 540
380 REM ** COMPUTER BLOKKEERT **
390 R = 9 : Y = 9
400 GOSUB 930
410 IF P <> 0 THEN 540
420 REM ** NIEUWE OPENING **
430 R = 1 : Y = 0
440 GOSUB 930
450 IF P <> 0 THEN 540
460 R = 0 : Y = 1
470 GOSUB 930
480 IF P <> 0 THEN 540
490 REM ** EERSTE DE BESTE VRIJE PLAATS **
500 FOR I = 1 TO 9
510 IF A(I) <> 0 THEN 530
520 P = I : I = 9
530 NEXT I
```

```
540 A(P) = 1
550 R = 1
560 GOSUB 1060
570 RETURN
580 :
590 REM *** UITVOER SPEELBORD ***
600 CLS
610 PRINT "*** BOTER KAAS EN EIEREN ***"
620 PRINT
630 IF C = 0 OR G = 0 THEN 660
640 PRINT "COMPUTER ZET OP VELD ";P
650 PRINT
660 PRINT A$
670 FOR I = 0 TO 2
680 FOR J = 1 TO 3
690 P = I * 3 + J
700 B$ = STR$(P)
710 IF A(P) = 0 THEN 750
720 B$ = " O"
730 IF A(P) = 1 THEN 750
740 B$ = " X"
750 PRINT TAB(J * 4 - 3)B$;
760 NEXT J
770 PRINT
780 PRINT A$
790 NEXT I
800 RETURN
810 :
820 REM *** ZET SPELER ***
830 PRINT "WELK VELD KIES JE ";
840 INPUT K
850 IF K < 1 OR K > 9 THEN 830
860 IF A(K) <> 0 THEN 830
870 A(K) = 9
880 B = B + 1 : G = G - 1 : R = 9
890 GOSUB 1060
900 RETURN
910 :
920 REM *** SCANROUTINE ***
930 RESTORE
940 FOR I = 1 TO 8
950 READ B(1),B(2),B(3)
960 FOR J = 1 TO 3
970 IF A(B(1)) <> 0 THEN 1000
980 IF A(B(2)) <> R OR A(B(3)) <> Y THEN 1000
990 P = B(1) : J = 3 : I = 8
1000 X = B(1) : B(1) = B(2)
1010 B(2) = B(3) : B(3) = X
1020 NEXT J
1030 NEXT I
1040 RETURN
1050 REM *** WINNAAR ? ***
1060 RESTORE
1070 FOR I = 1 TO 8
1080 READ X,Y,Z
```

```
1090 IF A(X) <> R THEN 1120
1100 IF A(Y) <> R OR A(Z) <> R THEN 1120
1110 F = R : G = Ø : I = 8
1120 NEXT I
1130 RETURN
1140 :
1150 REM *** WIE BEGINT ? ***
1160 PRINT "WIL JIJ BEGINNEN <JA/NEE> ";
1170 INPUT B$
1180 B$ = LEFT$(B$,1)
1190 H = 1
1200 IF B$ = "N" THEN 1230
1210 IF B$ <> "J" THEN 1160
1220 H = Ø
1230 G = G - H
1240 RETURN
1250 :
1260 REM *** RESULTAAT ***
1270 IF F = Ø THEN B$ = "GELIJK SPEL"
1280 IF F = 1 THEN B$ = "IK HEB GEWONNEN"
1290 IF F = 9 THEN B$ = "JIJ HEBT GEWONNEN"
1300 PRINT B$
1310 RETURN
1320 :
1330 REM **** STUURMODULE ****
1340 REM *** INITIALISEREN ***
1350 GOSUB 80
1360 REM *** UITVOER SPEELBORD ***
1370 GOSUB 600
1380 REM *** WIE BEGINT ? ***
1390 GOSUB 1160
1400 IF H = 1 THEN 1470
1410 REM *** UITVOER SPEELBORD ***
1420 GOSUB 600
1430 REM *** ZET SPELER ***
1440 GOSUB 830
1450 IF G <= Ø THEN 1500
1460 REM *** TEGENZET COMPUTER ***
1470 GOSUB 180
1480 IF F = Ø AND G > Ø THEN 1420
1490 REM *** RESULTAAT ***
1500 C = Ø
1510 GOSUB 600
1520 GOSUB 1270
1530 END
```

Voorbeeld

```
*** BOTER KAAS EN EIEREN ***

!---!---!---!
  1   2   3
!---!---!---!
  4   5   6
!---!---!---!
  7   8   9
!---!---!---!
WIL JIJ BEGINNEN <JA/NEE> JA


*** BOTER KAAS EN EIEREN ***

!---!---!---!
  1   2   3
!---!---!---!
  4   5   6
!---!---!---!
  7   8   9
!---!---!---!
WELK VELD KIES JE  5


*** BOTER KAAS EN EIEREN ***

COMPUTER ZET OP VELD  9

!---!---!---!
  1   2   3
!---!---!---!
  4   X   6
!---!---!---!
  7   8   O
!---!---!---!
WELK VELD KIES JE  1


*** BOTER KAAS EN EIEREN ***

COMPUTER ZET OP VELD  3

!---!---!---!
  X   2   O
!---!---!---!
  4   X   6
!---!---!---!
  7   8   O
!---!---!---!
WELK VELD KIES JE  6
```

```
*** BOTER KAAS EN EIEREN ***

COMPUTER ZET OP VELD  4

!---!---!---!
  X   2   O
!---!---!---!
  O   X   X
!---!---!---!
  7   8   O
!---!---!---!
WELK VELD KIES JE  8

*** BOTER KAAS EN EIEREN ***

COMPUTER ZET OP VELD  2

!---!---!---!
  X   O   O
!---!---!---!
  O   X   X
!---!---!---!
  7   X   O
!---!---!---!
WELK VELD KIES JE  7


*** BOTER KAAS EN EIEREN ***

!---!---!---!
  X   O   O
!---!---!---!
  O   X   X
!---!---!---!
  X   X   O
!---!---!---!
GELIJK SPEL
```

3.5 Doolhof

Het blijkt dat doolhoven op mensen een intrigerende aantrekkingskracht hebben. In veel tijdschriften komen we ze tegen en we zijn dan gauw geneigd onze pen te pakken en de oplossing te zoeken.
In de Griekse mythologie wordt ook al gewag gemaakt van een doolhof. Dit doolhof, waarin de beruchte Minotaurus was opgesloten, lag op het eiland Kreta. Het was ontworpen door Daidalos, dit is ook degene die met zijn zoon Ikaros vluchtte uit zijn gevangenschap met behulp van vleugels. Niet alleen in de Griekse oudheid komen doolhoven voor, maar ook in Engeland. Bekend is het heggendoolhof van Hampton Court. Om in de behoefte aan doolhoven tegemoet te komen is het volgende programma ontworpen. Het genereert steeds weer een ander doolhof om opgelost te worden. Elk doolhof heeft ten minste één oplossing, maar er kunnen er meer zijn. Beschikt men over een printer dan kan men het doolhof hierop uitprinten. De enige invoer die men moet geven is de gewenste hoogte en breedte van het doolhof. Het programma beperkt de breedte tot 30 en de hoogte tot 31 eenheden. Dit dient men wel aan de eigen computer aan te passen.

Dit programma is ook goed te gebruiken om aan de hand van het doolhof een driedimensionale weergave te maken. Men zal dan wel gebruik moeten maken van POKE-opdrachten om de plaatjes op te bouwen.
Een andere leuke uitbreiding is, de computer zijn weg te laten zoeken door het doolhof. De uitbreiding zal niet eenvoudig zijn, maar het is mogelijk.

Programma

```
10 REM *** DOOLHOF ***
20 DIM A(31,30), B(16)
30 RANDOM                         :REM DIT STATEMENT MAG VERVALLEN
40 GOTO 1040
50 :
60 REM *** INITIALISEREN ***
70 B$(1) = "   !"
80 B$(2) = "---!"
90 B$(3) = "    "
100 RETURN
110 :
120 REM *** INVOER ***
130 CLS                         :REM SCHERM WISSEN
140 PRINT "*** DOOLHOF ***"
150 PRINT
160 PRINT "HOOGTE VAN HET DOOLHOF ";
170 INPUT H
180 IF H < 1 OR H > 31 THEN 160
190 IF H/2 - INT(H/2) <> 0 THEN 210
200 H = H - 1
210 PRINT "BREEDTE VAN HET DOOLHOF ";
220 INPUT B
230 IF B < 0 OR B > 30 THEN 210
240 RETURN
250 :
260 REM *** DOOLHOF OPBOUWEN ***
270 REM ** BASIS DOOLHOF OPBOUWEN **
280 K = INT(B/3)
290 IF K < 6 THEN 310
300 K = 5
310 FOR X = 1 TO H STEP 2
320 FOR Y = 0 TO 1
330 I = X + Y
340 IF I > H THEN 380
350 FOR J = 1 TO B
360 A(I,J) = ABS(Y-1)
370 NEXT J
380 NEXT Y
390 NEXT X
400 REM ** DOORGANGEN MAKEN **
410 FOR I = 1 TO H STEP 2
420 X = INT(RND(0) * B + 1)
430 P = (I + 1)/2
440 IF I = 1 THEN 460
450 IF ABS(B(P-1) - X) > K THEN 420
460 B(P) = X
470 A(I,X) = 0
480 NEXT I
490 REM ** WILLEKEURIGE DOORGANGEN **
500 REM ** EXTRA INGANG ! **
510 X = INT(RND(0) * B + 1)
520 A(1,X) = 0
530 FOR I = 3 TO H - 2 STEP 2
```

```
540 Y = INT(RND(0) * K + 1)
550 FOR P = 1 TO Y
560 J = INT(RND(0) * B + 1)
570 IF A(I,J) = 0 THEN 560
580 A(I,J) = 0                          :REM DOORGANG
590 NEXT P
600 NEXT I
610 FOR I = 2 TO H-1 STEP 2
620 Y = INT(RND(0) * B + 1)
630 FOR P = 1 TO Y
640 J = INT(RND(0) * B + 1)
650 IF A(I,J) = 2 THEN 640
660 A(I,J) = 2
670 NEXT P
680 NEXT I
690 REM ** OBSTAKELS WEGHALEN **
700 FOR I = 2 TO H-1 STEP 2
710 P = (I + 2) / 2
720 A = B(P - 1)
730 C = B(P)
740 S = 1                    :REM STAPGROOTTE
750 IF A <= C THEN 770
760 S = -1
770 FOR J = A TO C STEP S
780 A(I,J) = 2
790 NEXT J
800 NEXT I
810 RETURN
820 :
830 REM *** UITVOER VAN DOOLHOF ***
840 CLS                      :REM SCHERM WISSEN
850 PRINT "*** DOOLHOF ***"
860 PRINT
870 FOR I = 1 TO H
880 PRINT "!";
890 FOR J = 1 TO B
900 P = A(I,J) + 1
910 PRINT B$(P);
920 IF A(I,J) <> 2 THEN 970
930 B$ = " "
940 IF J < B THEN 960
950 B$ = "!"
960 PRINT B$;
970 NEXT J
980 PRINT
990 NEXT I
1000 RETURN
1010 :
1020 REM *** STUURMODULE ***
1030 REM *** INITIALISEREN ***
1040 GOSUB 70
1050 REM *** INVOER ***
1060 GOSUB 130
1070 REM *** DOOLHOF OPBOUWEN ***
1080 GOSUB 280
```

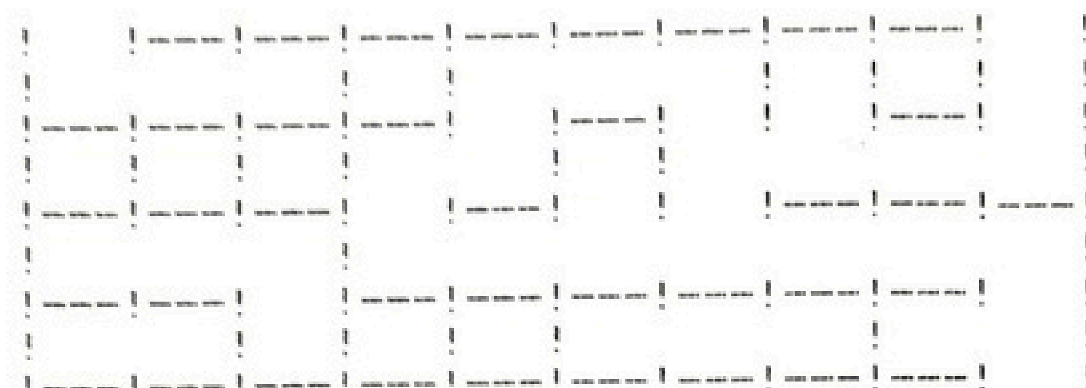
```
1090 REM *** UITVOER ***
1100 GOSUB 840
1110 PRINT "WILT U NOG EEN DOOLHOF <JA/NEE> ";
1120 INPUT B$
1130 IF LEFT$(B$,1) = "J" THEN 1040
1140 END
```

Voorbeeld

```
*** DOOLHOF ***

HOOGTE VAN HET DOOLHOF  10
BREEDTE VAN HET DOOLHOF  10


*** DOOLHOF ***
```

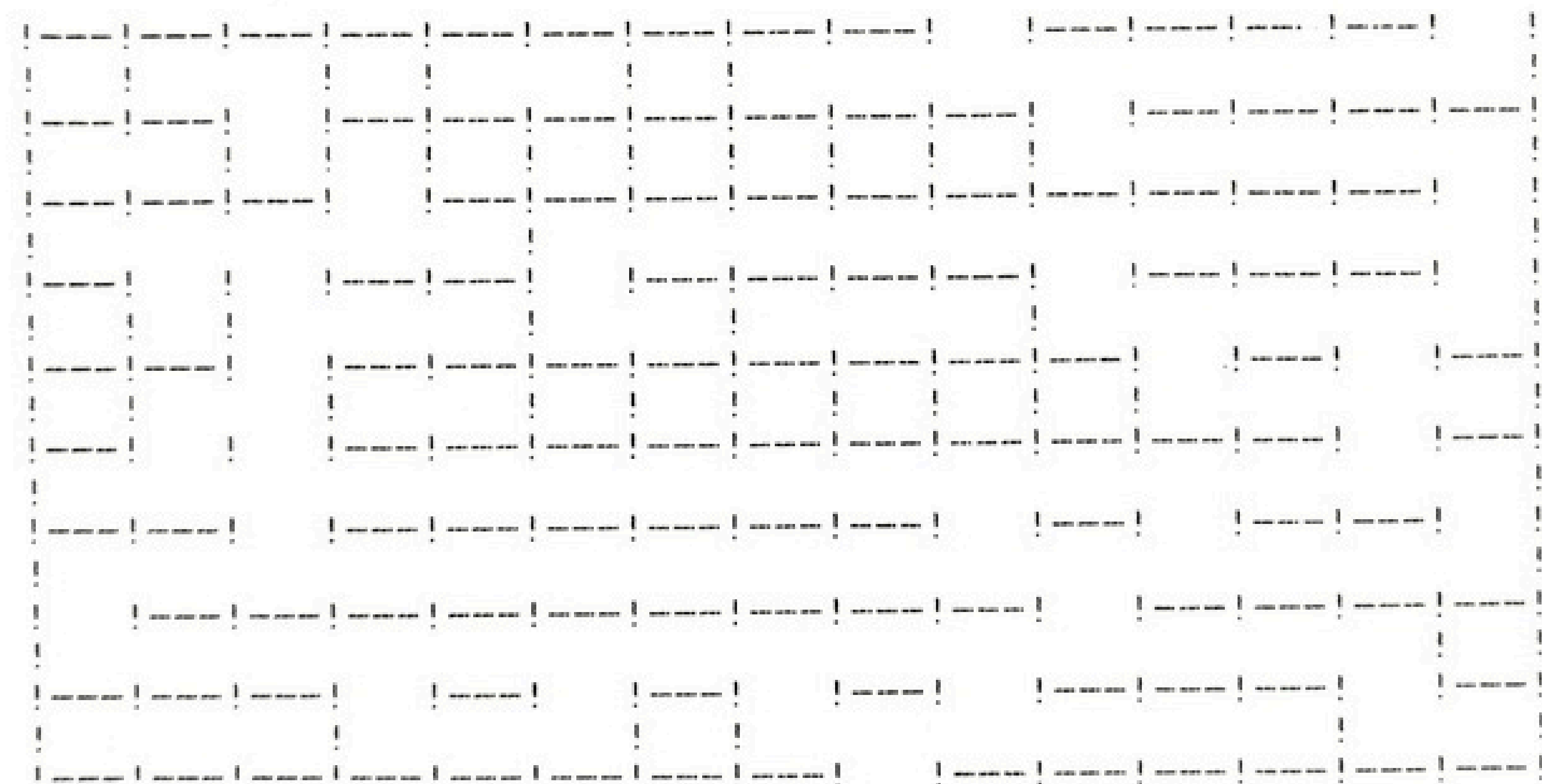


```
WILT U NOG EEN DOOLHOF <JA/NEE> JA

*** DOOLHOF ***

HOOGTE VAN HET DOOLHOF  20
BREEDTE VAN HET DOOLHOF  15


*** DOOLHOF ***
```



```
WILT U NOG EEN DOOLHOF <JA/NEE> NEE
```

3.6 Duikboot

In de Zuidelijke ijszee dichtbij Young's eiland is een vijandelijke onderzeeër door een satelliet gesignaleerd. Door de ruwe zee en de verschillende zoutlagen kan de positie niet nauwkeurig worden bepaald. Gelukkig is er een nieuw soort torpedojager in de buurt die ook nog een helikopter aan boord heeft met de nodige opsporingsapparatuur. Eén van de bijzonderheden van de torpedojager is dat deze onzichtbaar is voor de duikboot.
De helikopter, die aan boord is van de torpedojager, heeft een actieradius van minstens 200 mijl. Is de helikopter eenmaal in de lucht dan kan hij naar elke coördinaat worden gedirigeerd. Daar laat de helikopter zijn 'dipper' zakken en bepaalt zo de afstand tot de duikboot. De gegevens van de helikopter worden direct aan de torpedojager doorgeseind. Dezelfde soort apparatuur is aan boord van de torpedojager. De nauwkeurigheid van de afstandsbepaling neemt toe naarmate men dichter bij de duikboot is; dat is ook logisch omdat de zoutlagen dan minder invloed krijgen.
De torpedojager heeft zelf twee torpedo's aan boord die hij kan afvuren op de duikboot. Deze moeten wel binnen een afstand van vijf mijl van de duikboot worden afgevuurd. De helikopter kan maar vijf achtereenvolgende beurten in de lucht blijven. Dan moet hij terug naar het moederschip om te worden bijgetankt. Door de zware zee kan de torpedojager maar maximaal negen mijl per beurt worden verplaatst. Dit betekent maximaal vijf mijl in de X-richting en vijf mijl in de Y-richting.
Op het scherm van de operatieleider kunnen de verrichtingen globaal worden gevolgd. De torpedojager wordt op het scherm weergegeven met een T en de helikopter, wanneer hij in de lucht is, met een H.
Bovenin het beeldscherm worden achtereenvolgens de afstand van de duikboot tot de torpedojager gegeven, de coördinaten waarop de torpedojager zich bevindt en het aantal torpedo's dat de torpedojager aan boord heeft. Komt de helikopter in de lucht dan worden de gegevens van de helikopter ook op het beeldscherm vermeld.
Uit de gegevens van de satelliet valt af te leiden dat er iets aan de hand moet zijn met de duikboot. Hij schijnt heel langzaam te varen. Overigens ligt er op de tafel van de operatieleider een liniaal, ruitjespapier en een passer.

Beschikt men over een computer, waarop men zelf karakters kan definiëren, dan is het veel aardiger de torpedojager in het beeldscherm weer te geven met een bootje en de helikopter met een plaatje van een heli in plaats van de letters T en H.
Nadat de spelsituatie is gewijzigd, wordt het hele beeld gewist en opnieuw opgebouwd. Dit duurt altijd erg lang. Men kan dit versnellen door met POKE-opdrachten de verandering direct op het beeldscherm aan te brengen.

Programma

```
10 REM *** DUIKBOOT ***
20 RANDOM             :REM DIT STATEMENT MAG VERVALLEN
30 GOTO 1800
40 :
50 REM *** INITIALISEREN ***
60 W1 = INT(RND(0) * 90) : REM X-COORD. TORPEDOJAGER
70 W2 = INT(RND(0) * 90) : REM Y-COORD. TORPEDOJAGER
80 D1 = INT(RND(0) * 90) : REM X-COORD. DUIKBOOT
90 D2 = INT(RND(0) * 90) : REM Y-COORD. DUIKBOOT
100 A = SQR((W1 - D1)↑2 + (W2 - D2)↑2)
```

```
110 IF A < 25 THEN 60
120 H3 = 5                    :REM HELI VOL GETANKT
130 R = 1                     :REM DUIKBOOT VAART
140 Z = 9                     :REM HELI AAN BOORD
150 C$ = ""                   :REM LEGE TEKST STRING
160 T = 2                     :REM AANTAL TORPEDO'S
170 E = 1                     :REM SPEL DRAAIT
180 RETURN
190 :
200 REM *** INVOER ***
210 PRINT "1 = HELI   ";
220 PRINT "2 = SCHIP  ";
230 PRINT "3 = TORPEDO"
240 PRINT "WAT IS UW KEUZE ";
250 INPUT X
260 IF X < 1 OR X > 3 THEN 210
270 ON X GOSUB 310  , 430  , 530
280 RETURN
290 :
300 REM *** INVOER/VERWERKING HELI ***
310 IF Z = 0 THEN 400
320 PRINT "GEEF X,Y-COORD. HELI ";
330 INPUT H1,H2
340 IF H1 < 0 OR H1 > 100 THEN 310
350 IF H2 < 0 OR H2 > 100 THEN 310
360 Z = 1                      :REM HELI IN DE LUCHT
370 IF H1 <> W1 OR H2 <> W2 THEN 400
380 H3 = 5                    :REM HELI WORDT BIJGETANKT
390 Z = 9                     :REM HELI AAN BOORD
400 RETURN
410 :
420 REM *** INVOER/VERWERKING TORPEDOJAGER ***
430 PRINT "GEEF X,Y-COORD TORPEDOJAGER ";
440 INPUT X,Y
450 IF ABS(W1 - X) > 5 OR ABS(W2 - Y) > 5 THEN 430
460 IF X < 0 OR X > 100 THEN 430
470 IF Y < 0 OR Y > 100 THEN 430
480 W1 = X
490 W2 = Y
500 RETURN
510 :
520 REM *** INVOER / VERWERKING TORPEDO ***
530 PRINT "GEEF DE RICHTING IN GRADEN ";
540 INPUT B
550 IF B < 0 OR B > 360 THEN 530
560 B = B * 3.1415926 / 180     :REM GRADEN => RAD
570 FOR I = 1 TO 5 STEP 0.2
580 X1 = W1 + COS(B) * I
590 X2 = W2 + SIN(B) * I
600 A = SQR((D1 - X1)↑2 + (D2 - X2)↑2)
610 IF A > 5 THEN 640
620 R = 0                     :REM DUIKBOOT GEZONKEN
630 I = 5                     :REM LUS-TELLER MAXIMUM
640 NEXT I
650 E = 0
```

```
660 T = T - 1                    :REM TORPEDO'S MIN EEN
670 IF R = 0 OR T = 0 THEN 720
680 E = 1
690 REM ** DUIKBOOT GEVLUCHT **
700 D1 = INT(RND(0) * D1 + 1)
710 D2 = INT(RND(0) * D2 + 1)
720 RETURN
730 :
740 REM *** KOERS DUIKBOOT ***
750 IF E = 0 THEN 860
760 D1 = D1 + INT(RND(0) * 3 - 1)
770 D2 = D2 + INT(RND(0) * 3 - 1)
780 IF D1 > -1 THEN 800
790 D1 = 90
800 IF D1 < 91 THEN 820
810 D1 = 0
820 IF D2 > -1 THEN 840
830 D2 = 90
840 IF D2 < 91 THEN 860
850 D2 = 0
860 RETURN
870 :
880 REM *** BEREKENING ***
890 REM ** HELICOPTER IN DE LUCHT **
900 IF Z <> 1 OR E = 0 THEN 980
910 H3 = H3 - 1                          :REM BRANDSTOF -1
920 IF H3 > -1 THEN 980
930 Z = 0
940 C$ = " NEERGESTORT !"
950 H1 = 0
960 H2 = 0
970 REM ** PLAATS HELICOPTER IN KAART **
980 H4 = -1                              :REM AANDUIDING HELI
990 IF Z <> 1 THEN 1020
1000 H4 = INT((H2 + 5)/10) * 10 + INT((H1 + 5)/10)
1010 REM ** PLAATS TORPEDOJAGER IN KAART **
1020 W4 = INT((W2 + 5)/10) * 10 + INT((W1 + 5)/10)
1030 REM ** PLAATS DUIKBOOT IN KAART **
1040 D4 = INT(( D2 + 5)/10) * 10 + INT((D1 + 5)/10)
1050 D$ = "*"                            :REM GEZONKEN DUIKBOOT
1060 IF R = 0 THEN 1110
1070 D$ = "D"                            :REM VARENDE DUIKBOOT
1080 IF E = 0 THEN 1110
1090 D4 = -1                   :REM DUIKBOOT ONZICHTBAAR
1100 REM ** AFSTAND TOT DUIKBOOT **
1110 H = 0
1120 IF Z <> 1 THEN 1180
1130 REM * HELI - DUIKBOOT *
1140 X = SQR((D1 - H1)↑2 + (D2 - H2)↑2)
1150 GOSUB 1240
1160 H = X
1170 REM * TORPEDOJAGER - DUIKBOOT *
1180 X = SQR((D1 - W1)↑2 + (D2 - W2)↑2)
1190 GOSUB 1240
1200 W = X
```

```
1210 RETURN
1220 :
1230 REM *** BEPALING MATE VAN NAUWKEURIGHEID ***
1240 X1 = INT(X * 10 + 0.5)/ 10
1250 IF X < 10 THEN 1290
1260 X1 = (INT(X / 5)) * 5
1270 IF X < 25 THEN 1290
1280 X1 = (INT(X / 10)) * 10
1290 X = X1
1300 RETURN
1310 :
1320 REM *** UITVOER ***
1330 CLS                              :REM SCHERM WISSEN
1340 PRINT "*** DUIKBOOT ***"
1350 IF Z = 9 THEN 1370
1360 PRINT "HELICOPTER ";H;TAB(17)"COORD.X-Y";H1;H2;C$
1370 PRINT "TORPEDOJAGER";W;TAB(17)"COORD.X-Y";W1;W2
1380 PRINT "TORP.";T
1390 FOR J = 0 TO 9
1400 PRINT TAB(J * 3 + 3) J;
1410 NEXT J
1420 PRINT " (* 10)"
1430 FOR I = 0 TO 9
1440 PRINT I*10;
1450 FOR J = 0 TO 9
1460 P = I * 10 + J
1470 A$ = "."
1480 IF P <> H4 THEN 1500
1490 A$ = "H"
1500 IF P <> W4 THEN 1520
1510 A$ = "T"
1520 IF P <> D4 THEN 1560
1530 A$ = D$
1540 IF P <> W4 OR P <> D4 THEN 1560
1550 A$ = "T" + D$
1560 IF P <> W4 OR P <> H4 THEN 1580
1570 A$ = "TH"
1580 IF P <> H4 OR P <> D4 THEN 1600
1590 A$ = "H" + D$
1600 PRINT TAB(J * 3 + 4)A$;
1610 NEXT J
1620 PRINT
1630 NEXT I
1640 RETURN
1650 :
1660 REM *** RESULTAAT ***
1670 REM ** BEREKENING **
1680 GOSUB 900
1690 REM ** UITVOER **
1700 GOSUB 1330
1710 D$ = "VERLOREN, DE DUIKBOOT VAART NOG STEEDS !"
1720 IF R = 1 THEN 1740
1730 D$ = "GEWONNEN, DE DUIKBOOT IS GEZONKEN !"
1740 PRINT D$
1750 PRINT "COORDINATEN DUIKBOOT ";D1;",";D2
```

```
1760 RETURN
1770 :
1780 REM **** STUURMODULE ****
1790 REM *** INITIALISEREN ***
1800 GOSUB 60
1810 REM *** BEREKENING ***
1820 GOSUB 900
1830 REM *** UITVOER ***
1840 GOSUB 1330
1850 REM *** INVOER ***
1860 GOSUB 210
1870 REM *** KOERS DUIKBOOT ***
1880 GOSUB 750
1890 IF E = 1 THEN 1820
1900 REM *** RESULTAAT ***
1910 GOSUB 1680
1920 END
```

Voorbeeld

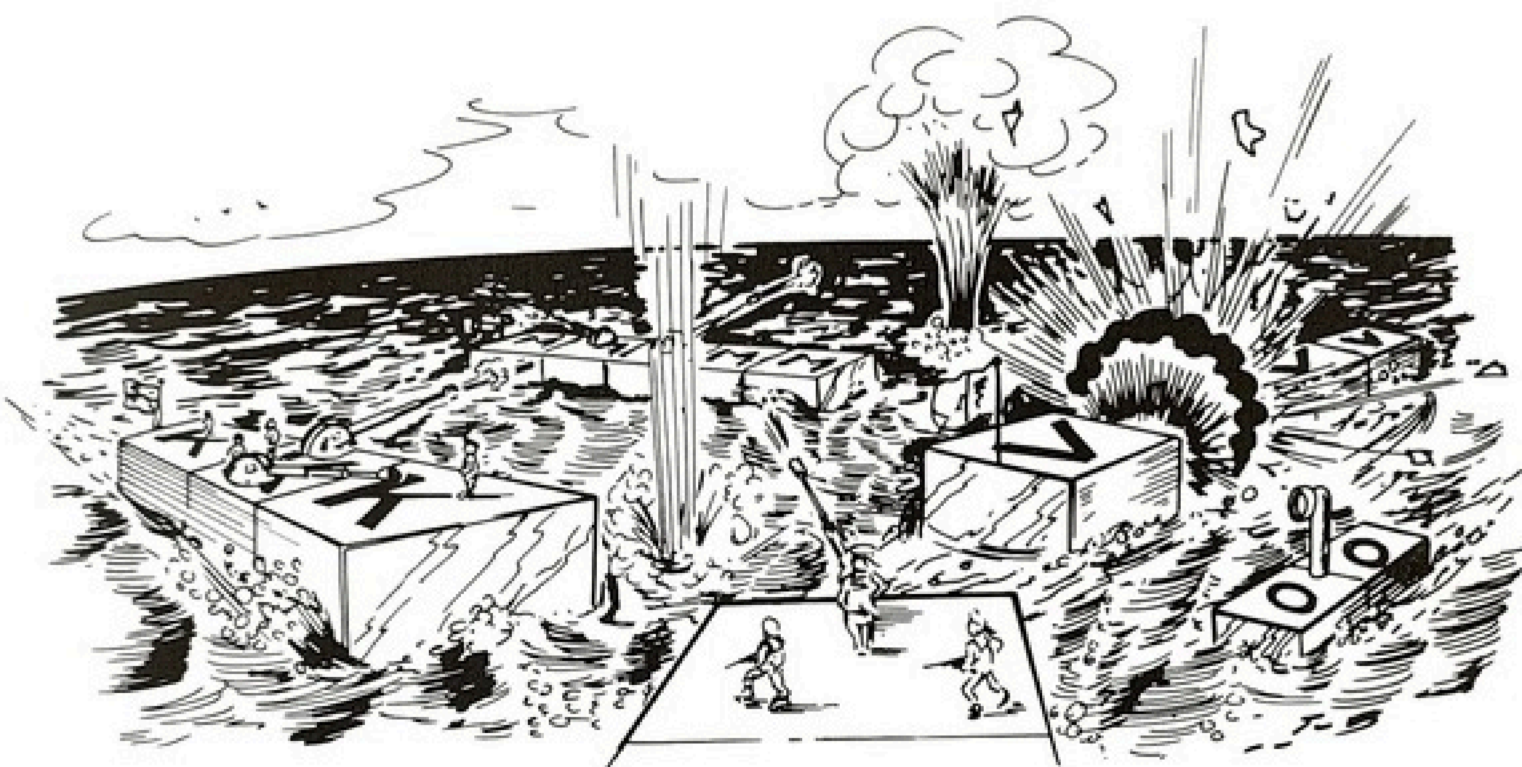
```
*** DUIKBOOT ***
TORPEDOJAGER 60   COORD.X-Y 44   14 TORP. 2
     0  1  2  3  4  5  6  7  8  9  (* 10)
  0  .  .  .  .  .  .  .  .  .  .
 10  .  .  .  .  T  .  .  .  .  .
 20  .  .  .  .  .  .  .  .  .  .
 30  .  .  .  .  .  .  .  .  .  .
 40  .  .  .  .  .  .  .  .  .  .
 50  .  .  .  .  .  .  .  .  .  .
 60  .  .  .  .  .  .  .  .  .  .
 70  .  .  .  .  .  .  .  .  .  .
 80  .  .  .  .  .  .  .  .  .  .
 90  .  .  .  .  .  .  .  .  .  .
1 = HELI   2 = SCHIP   3 = TORPEDO
WAT IS UW KEUZE  1
GEEF X,Y-COORD. HELI  30 , 60
```

```
*** DUIKBOOT ***
HELICOPTER  10    COORD.X-Y 30   60
TORPEDOJAGER 60   COORD.X-Y 44   14 TORP. 2
     0  1  2  3  4  5  6  7  8  9  (* 10)
  0  .  .  .  .  .  .  .  .  .  .
 10  .  .  .  .  T  .  .  .  .  .
 20  .  .  .  .  .  .  .  .  .  .
 30  .  .  .  .  .  .  .  .  .  .
 40  .  .  .  .  .  .  .  .  .  .
 50  .  .  .  .  .  .  .  .  .  .
 60  .  .  .  H  .  .  .  .  .  .
 70  .  .  .  .  .  .  .  .  .  .
 80  .  .  .  .  .  .  .  .  .  .
 90  .  .  .  .  .  .  .  .  .  .
1 = HELI   2 = SCHIP   3 = TORPEDO
WAT IS UW KEUZE  1
GEEF X,Y-COORD. HELI  33 , 60

*** DUIKBOOT ***
HELICOPTER  10    COORD.X-Y 33   60
TORPEDOJAGER 60   COORD.X-Y 44   14 TORP. 2
     0  1  2  3  4  5  6  7  8  9  (* 10)
  0  .  .  .  .  .  .  .  .  .  .
 10  .  .  .  .  T  .  .  .  .  .
 20  .  .  .  .  .  .  .  .  .  .
 30  .  .  .  .  .  .  .  .  .  .
 40  .  .  .  .  .  .  .  .  .  .
 50  .  .  .  .  .  .  .  .  .  .
 60  .  .  .  H  .  .  .  .  .  .
 70  .  .  .  .  .  .  .  .  .  .
 80  .  .  .  .  .  .  .  .  .  .
 90  .  .  .  .  .  .  .  .  .  .
1 = HELI   2 = SCHIP   3 = TORPEDO
WAT IS UW KEUZE
```

3.7 Zeeslag

De spelregels van Zeeslag zijn als volgt: men formeert in een zee zijn vloot. De vloot bestaat uit een vliegdekschip, die vijf posities inneemt, een kruiser met vier posities, een fregat met drie posities, een mijnenveger met twee posities en een duikboot met twee posities. Deze schepen moeten minstens één positie van elkaar afliggen. Verder mogen de schepen in de horizontale (west-oost) en de verticale (noord-zuid) richting liggen. De computer bouwt ook zijn vloot en houdt zich ook aan bovengenoemde regels. Wanneer de speler zijn vloot opbouwt, dient hij er rekening mee te houden dat er gerekend wordt vanaf de boeg van het schip.
Nadat de vloten geformeerd zijn, wordt met schieten begonnen. Men geeft de coördinaten van het schot in de X- en Y-richting. Is het schot mis dan verschijnt er 'PLONS' op het scherm en tegelijkertijd wordt het schot in de zee aangegeven. Is het schot raak dan wordt aangegeven wat men raak geschoten heeft.



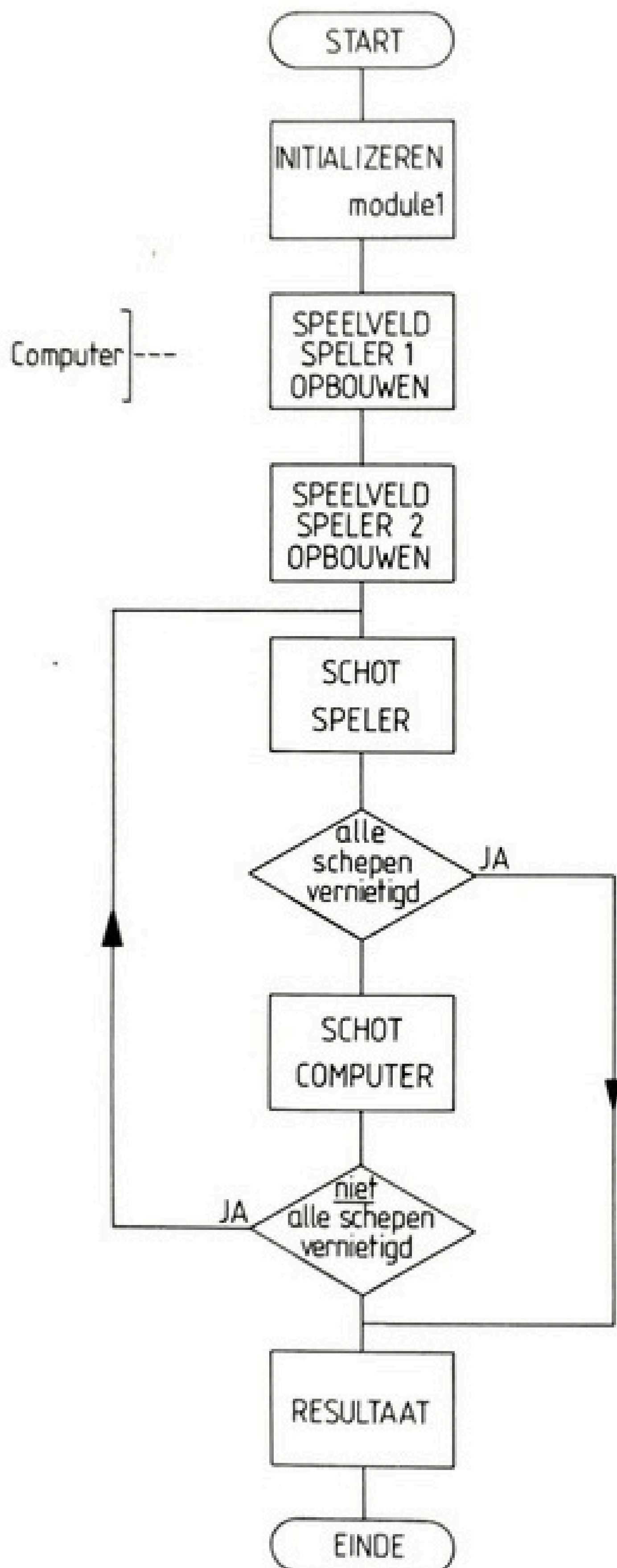
Eén van de nadelen van het programma is dat men steeds moet wachten, totdat het hele beeldscherm is opgebouwd. Men kan dit voorkomen door de nieuw ontstane spelsituatie op het beeldscherm te POKEn.
Beschikt men over een kleurencomputer dan kan men de zee waar de vijandelijke vloot ligt een andere kleur geven dan de zee waar de eigen vloot ligt.
Het spel Zeeslag is een oud spel dat vroeger werd gespeeld op ruitjespapier maar tegenwoordig ook mogelijk is op een computer. Het spel wordt niet door twee personen gespeeld, maar men speelt tegen de computer.
De computer moet beslissingen nemen afhankelijk van de spelsituatie. Er is in dit geval gekozen voor een eenvoudige tactiek om het spel niet al te groot te maken. Op dit punt kan het spel aanmerkelijk worden verbeterd.

Het programma Zeeslag is een goed voorbeeld hoe een idee in een modulair programma is omgezet. Het spel is onder te verdelen in een aantal onderdelen:

1. Beide spelers nemen een stuk papier en zetten er een coördinatenstelsel op.
2. Speler 1 (computer) plaatst niet zichtbaar voor speler 2 zijn schepen in een coördinatenstelsel, waarbij hij er op let zich aan de spelregels te houden.

3. Speler 2 plaatst niet zichtbaar voor speler 1 zijn schepen, waarbij hij er op let dat hij zich aan de coördinaten houdt.
4. Speler 1 lost een schot in de zee van speler 2. Heeft speler 1 alle schepen van speler 2 vernietigd dan heeft hij gewonnen.
5. Speler 2 lost een schot in de zee van speler 1. Heeft speler 2 alle schepen van speler 1 vernietigd dan heeft hij gewonnen.
6. De winnaar wordt bekend gemaakt.

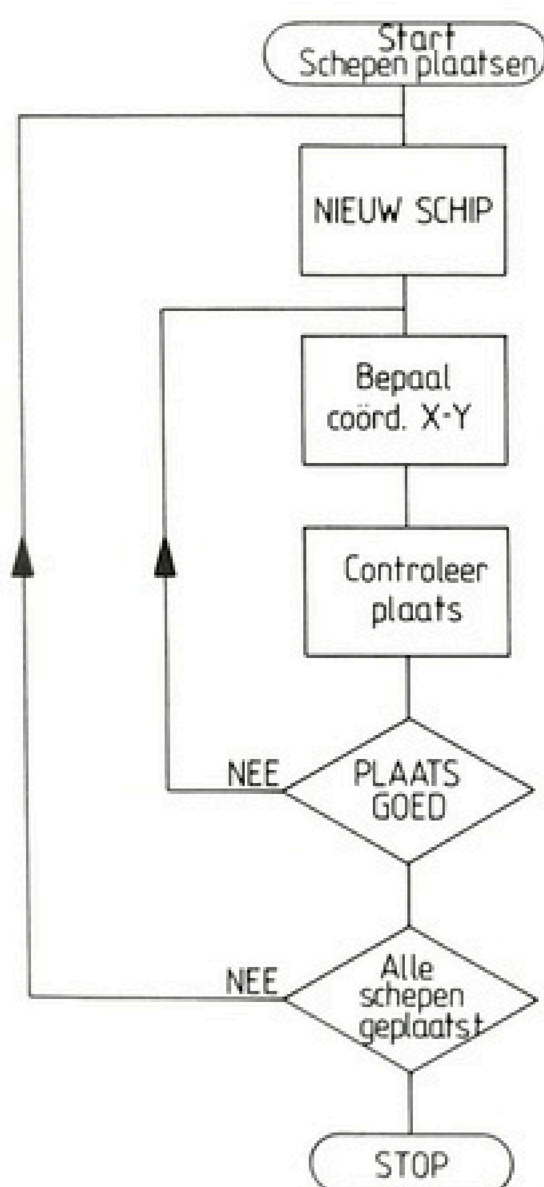
Wanneer het bovenstaande in een stroomschema wordt weergegeven dan ziet dit er als volgt uit:



Het bovenstaande schema is terug te vinden in de stuurmodule achteraan in het programma. De getekende rechthoeken stellen subroutines voor die de omschreven functie uitvoeren.
Het opbouwen van de speelvelden van zowel computer als speler geschiedt op identieke wijze.
Achtereenvolgens worden de vijf schepen geplaatst.

1. Er wordt een coördinaat bepaald waar de boeg van het schip komt te liggen. De computer bepaalt dit willekeurig.
2. Daarna wordt bepaald of het schip noord-zuid of west-oost komt te liggen.
3. Na de bovenstaande actie (punt 1 en 2) wordt nagegaan of het schip op de aangegeven plaats kan liggen. Voldoen de coördinaten aan de spelregels?
4. Kan het schip er niet liggen dan wordt er weer bij punt 1 begonnen. In het andere geval wordt het schip neergelegd.
5. Wanneer nog niet alle schepen zijn geplaatst, neem dan het volgende schip uit de rij en begin weer bij 1.

Schematisch ziet dit er als volgt uit:

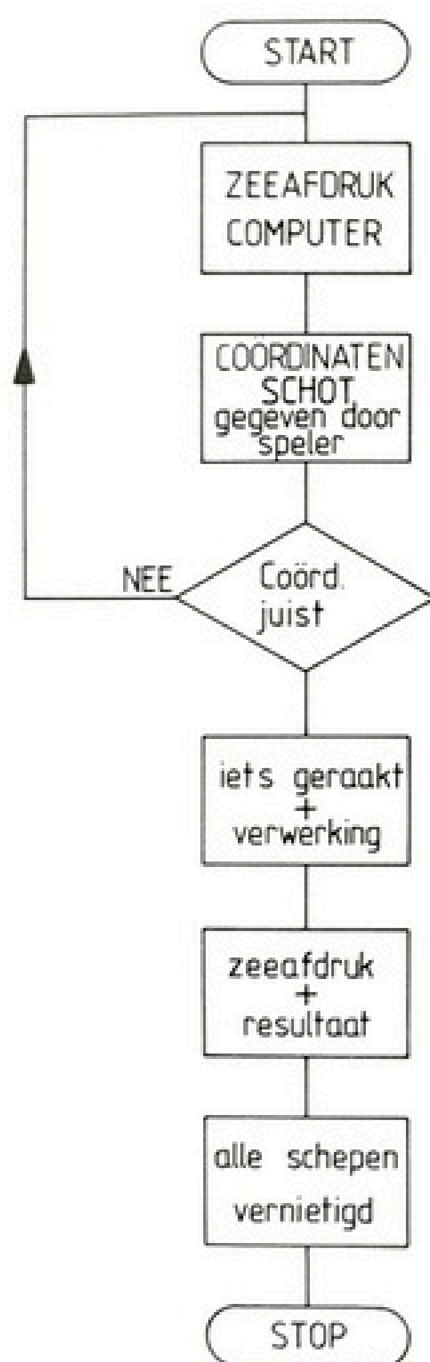


Het schema van de subroutine is terug te vinden in het programma, waar het twee keer voorkomt. De eerste keer, waar de computer zijn speelveld opbouwt en de tweede keer waar de speler zijn speelveld opbouwt. Wanneer men de subroutines '*** SPEELVELD COMPUTER OPBOUWEN ***' en '*** SPEELVELD SPELER OPBOUWEN ***' bekijkt, zal men zien dat beide routines gebruik maken van de subroutine '*** CONTROLE OP SCHIP ***'. Deze subroutine gaat na of de plaatsing van het schip juist is en of het voldoet

aan de spelregels. Worden de opgegeven coördinaten geaccepteerd, dan wordt het schip ook daadwerkelijk geplaatst. In het andere geval wordt de vraag opnieuw gesteld.
Nadat het programma deze modules heeft doorlopen, zijn alle schepen geplaatst in het coördinatenstelsel en het spel kan beginnen. De speler begint met schieten in de zee van de computer. De volgende functies worden verricht:

1. Zee opbouwen van de computer, zonder het weergeven van de schepen van de computer. Dit geschiedt met stuurvariabelen en de subroutine '*** ZEE-AFDRUK ***'.
2. Speler geeft de coördinaten van het schot in de X-Y-richting op.
3. Het programma controleert de coördinaten. Worden deze niet geaccepteerd, terug naar punt 1.
4. Nagegaan wordt of er iets geraakt is en wát er geraakt is. Het schot wordt verwerkt in de zee van de computer.
5. Resultaat van het schot wordt op het beeldscherm gezet.
6. Nagegaan wordt of alle schepen van de computer vernietigd zijn.

In stroomschema ziet het er als volgt uit:

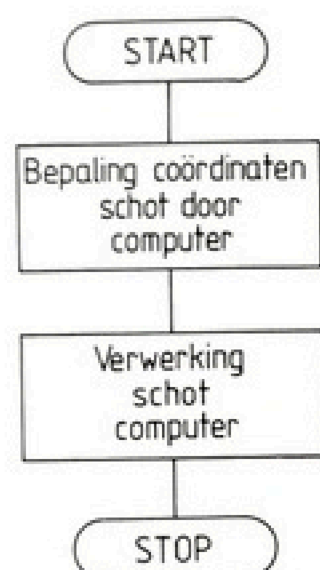


De module waarin het schot van de computer op de speler wordt bepaald, zit iets gecompliceerder in elkaar. Degenen die exact willen weten wat er gebeurt, kunnen het beste het

programma statement voor statement doorlopen. Hierna wordt slechts een globale uiteenzetting gegeven.
Bij de bepaling van het schot door de computer kan van drie situaties worden uitgegaan:

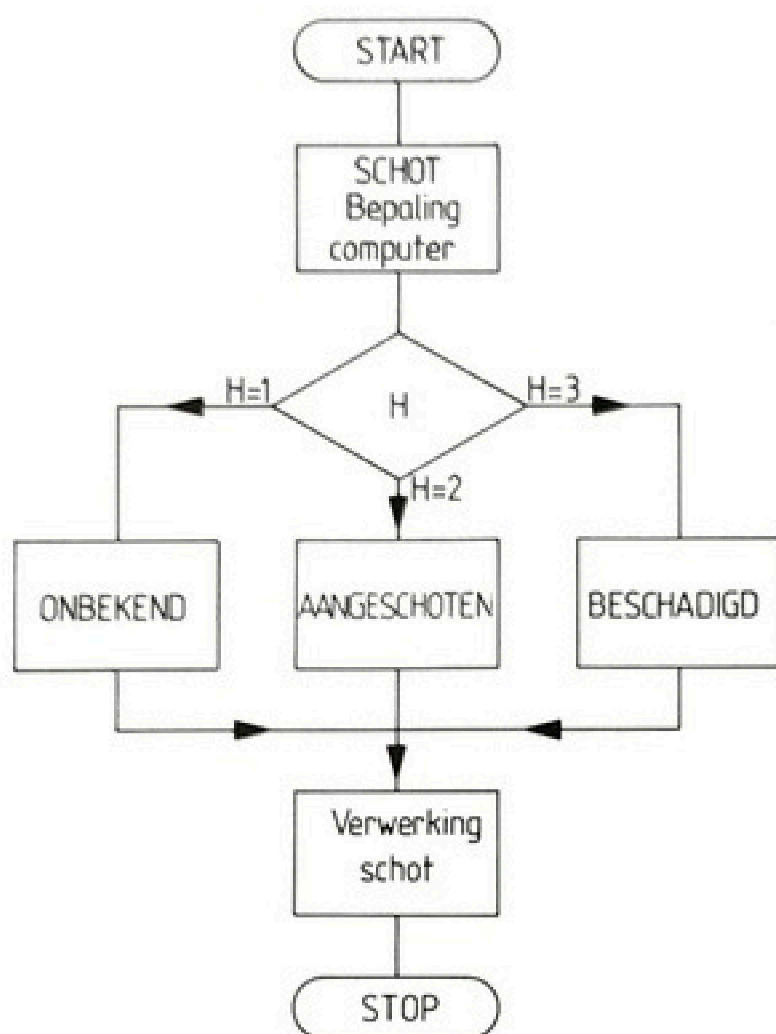
1. onbekend
2. aangeschoten
3. beschadigd

De drie genoemde opmerkingen zijn ook in de programmatuur terug te vinden.
Onbekend wil zeggen: de computer weet niet waar een schip ligt. Hij gokt!
Aangeschoten wil zeggen: Een schot is raak, maar hij weet nog niet hoe het schip ligt horizontaal of verticaal.
Beschadigd wil zeggen: Twee of meer schoten zijn raak. De richting waarin het schip ligt is bekend.
De module *** SCHOT COMPUTER *** is te verdelen in twee blokken. Het eerste blok bepaalt de coördinaten van het schot. Het tweede blok gaat na wat de resultaten van het schot zijn:



Het blok *** BEPALING COORDINATEN VAN HET SCHOT *** is opgebouwd uit een hoofdmodule met drie submodules. In de hoofdmodule wordt bepaald of een schip van de speler aangeschoten of beschadigd is. Blijkt dit voor geen enkel schip van de speler op te gaan dan geldt 'onbekend'. De bovenstaande bepaling geschiedt aan de hand van de inhoud van matrix B. De aansturing van de submodules geschiedt door de stuurvariabele H.

In schema ziet dit er als volgt uit:



De laatste module is de module '*** RESULTAAT ***'. Deze geeft aan wie de winnaar van het spel is.

Het zal duidelijk zijn dat het verdelen van het probleem 'Zeeslag' in deelproblemen, het oplossen eenvoudig maakt. Bij de ontwikkeling kan men zich volledig concentreren op een deelprobleem. Door dit deelprobleem in een module op te lossen, blijft het geheel overzichtelijk en voorkomt men spaghetti-toestanden. Tegelijkertijd heeft men het voordeel dat men een module al kan testen voordat het gehele programma klaar is.
Met deze uitleg is er naar gestreefd, de gebruiker van dit boek een inzicht te geven hoe complexe problemen kunnen worden aangepakt.

Programma

```
10 REM *** ZEESLAG ***
20 RANDOM                    :REM DIT STATEMENT KAN VERVALLEN
30 DIM A(20,10),B(5,3),A$(12),B$(5)
40 DATA 5,2, 4,3, 3,4, 3,5, 2,6
50 DATA " . ",-V-,-K-,-F-,-M-,-D-
60 DATA " * ",*V*,*K*,*F*,*M*,*D*
70 DATA VLIEGDEKSCHIP,KRUISER,FREGAT
80 DATA MIJNENVEGER,DUIKBOOT
90 GOTO 3440
100 :
110 REM *** INITIALISEREN ***
120 RESTORE                  :REM DATAPOINTER VOORAAN
130 FOR I = 1 TO 20
140 FOR J = 1 TO 10
150 IF I <> 1 THEN 170
160 READ X                   :REM DUMMY LEZEN
170 A(I,J) = 1               :REM START WAARDE
180 NEXT J
190 IF I > 12 THEN 240
200 READ A$(I)               :REM SYMBOLEN
210 IF I > 5 THEN 240
220 B(I,1) = 0
230 B(I,2) = 0
240 IF I < 16 THEN 260
250 READ B$(I - 15)          :REM NAMEN SCHEPEN
260 NEXT I
270 RESTORE
280 FOR I = 1 TO 5
290 READ B(I,3),X
300 NEXT I
310 REM ** VARIABELEN OP NUL **
320 T = 0                    :REM GEZONKEN SCHEPEN
330 RETURN
340 :
350 REM *** SPEELVELD COMPUTER OPBOUWEN ***
360 RESTORE
370 FOR X = 1 TO 5
380 READ Y,V                 :REM GROOTTE EN SOORT
390 Z = INT(RND(0) * (10 - Y) + 1)
400 L = INT(RND(0) * 10 + 1)
410 H = INT(RND(0) * 2 )
420 IF H = 1 THEN 470
430 REM ** WISSELEN VAN Z MET L **
440 M = Z
450 Z = L
460 L = M
470 Q = 0                   :REM DEEL 1 MAT A
480 GOSUB 540
490 IF M = 1 THEN 390
500 NEXT X
510 RETURN
520 :
530 REM *** CONTROLE OP SCHIP ***
```

```
540 R1 = Z - 1
550 R2 = Z + Y
560 K1 = L - 1
570 K2 = L + 1
580 IF H = 1 THEN 610
590 R2 = Z + 1
600 K2 = L + Y
610 M = 0                     :REM HULPVARIABELE
620 FOR I = R1 TO R2
630 FOR J = K1 TO K2
640 IF I < 1 OR J < 1 THEN 700
650 IF I > 10 OR J > 10 THEN 700
660 IF A(I + Q * 10,J) = 1 THEN 700
670 I = R2                    :REM LUS-TELLER OP MAXIMUM
680 J = K2                    :REM LUS-TELLER OP MAXIMUM
690 M = 1                     :REM FOUT
700 NEXT J
710 NEXT I
720 IF M = 1 THEN 890
730 REM ** SCHIP PLAATSEN **
740 R1 = Z
750 R2 = Z + Y - 1
760 K1 = L
770 K2 = L
780 IF H = 1 THEN 810
790 R2 = Z
800 K2 = L + Y  - 1
810 M = 1
820 IF K2 > 10 OR R2 > 10 THEN 890
830 FOR I = R1 TO R2
840 FOR J = K1 TO K2
850 A(I + Q * 10,J) = V
860 NEXT J
870 NEXT I
880 M = 0
890 RETURN
900 :
910 REM *** SPEELVELD SPELER OPBOUWEN ***
920 RESTORE
930 REM ** ZEE SPELER SCHOONVEGEN **
940 FOR I = 11 TO 20
950 FOR J = 1 TO 10
960 A(I,J) = 1
970 NEXT J
980 NEXT I
990 S = 1
1000 Q = 1                    :REM DEEL 2 MAT A
1010 FOR X = 1 TO 5
1020 READ Y,V
1030 D$ = " -VLOOTOPBOUW-"
1040 REM ** ZEE AFDRUKKEN **
1050 GOSUB 1300
1060 PRINT "GEEF X EN Y-COORD VAN ";B$(X);
1070 PRINT " LENGTE ";Y;
1080 INPUT L,Z
```

```
1090 IF L < 1 OR L > 10 THEN 1030
1100 IF Z < 1 OR Z > 10 THEN 1030
1110 PRINT "LIGGING NOORD-ZUID OF WEST-OOST ";
1120 INPUT A$
1130 A$ = LEFT$(A$,1)
1140 H = 1
1150 IF A$ = "N" THEN 1190
1160 H = 0
1170 IF A$ <> "W" THEN 1110
1180 REM ** CONTROLE OP SCHIP **
1190 GOSUB 540
1200 IF M = 1 THEN 1030
1210 NEXT X
1220 REM ** ZEE-AFDRUK **
1230 GOSUB 1300
1240 PRINT "NOG WIJZIGEN <JA/NEE> ";
1250 INPUT A$
1260 IF LEFT$(A$,1) = "J" THEN 920
1270 RETURN
1280 :
1290 REM *** ZEE-AFDRUK ***
1300 CLS                    :REM SCHERM WISSEN
1310 A$ = "- COMPUTER -"
1320 IF Q = 0 THEN 1340
1330 A$ = "- SPELER -"
1340 PRINT "*** ZEESLAG ***    ";D$
1350 PRINT TAB(15)A$
1360 IF K = 0 THEN 1380
1370 PRINT TAB(K * 3 - 2 + 5)"*"
1380 PRINT TAB(6)"1  2  3  4  5  6  ";
1390 PRINT "7  8  9  10"
1400 FOR I = 1 TO 10
1410 PRINT I;
1420 IF I <> R THEN 1440
1430 PRINT TAB(3)"*";
1440 PRINT TAB(5);
1450 P = I + Q * 10
1460 FOR J = 1 TO 10
1470 A$ = " . "
1480 IF A(P,J) = -1 THEN 1520
1490 IF Q = S THEN 1510
1500 IF A(P,J) < 7 THEN 1520
1510 A$ = A$(A(P,J))
1520 PRINT A$;
1530 NEXT J
1540 PRINT
1550 NEXT I
1560 RETURN
1570 :
1580 REM *** VERWERKING SCHOT COMPUTER ***
1590 GOSUB 1300                   :REM SPEELVELD
1600 X = A(R + Q*10,K) - 1
1610 A(R+Q*10 , K) = A(R+Q*10 , K) + 6
1620 PRINT "IS ER IETS GERAAKT <JA/NEE> ";
1630 INPUT D$
```

```
1640 IF LEFT$(D$,1) = "N" THEN 1660
1650 IF LEFT$(D$,1) <> "J" THEN 1620
1660 IF X = 0 THEN 1980
1670 RESTORE
1680 FOR I = 1 TO X
1690 READ Y,V
1700 NEXT I
1710 B(X,2) = B(X,2) + 1          :REM SCHOTEN SCHIP
1720 Z = R * 100 + K
1730 IF B(X,2) = 1 THEN 1750
1740 IF Z > B(X,1) THEN 1770
1750 B(X,1) = Z
1760 REM ** SCHIP UITGESCHAKELD ? **
1770 IF Y <> B(X,2) THEN 1980
1780 REM ** SCHIP GEZONKEN **
1790 REM * OMGEVING SCHIP MERKEN *
1800 R = INT(B(X,1)/100)
1810 K = B(X,1) - R * 100
1820 R2 = R + Y - 1
1830 K2 = K
1840 IF R2 > 10 THEN 1860
1850 IF A(R2+Q*10 , K2) = V + 6 THEN 1880
1860 R2 = R
1870 K2 = K + Y - 1
1880 FOR I = R - 1 TO R2 + 1
1890 FOR J = K - 1 TO K2 + 1
1900 IF I < 1 OR J < 1 THEN 1940
1910 IF I > 10 OR J > 10 THEN 1940
1920 IF A(I+Q*10 , J) <> 1 THEN 1940
1930 A(I+Q*10 , J) = -1
1940 NEXT J
1950 NEXT I
1960 B(X,2) = B(X,2) * -1
1970 T = T + 1          :REM TOTAAL GEZONKEN SCHEPEN
1980 RETURN
1990 :
2000 REM *** SCHOT SPELER ***
2010 Q = 0
2020 S = 1
2030 R = 0
2040 K = 0
2050 D$ = ""
2060 GOSUB 1300
2070 PRINT "GEEF X , Y - COORD ";
2080 INPUT K,R
2090 IF K < 1 OR K > 10 THEN 2070
2100 IF R < 1 OR R > 10 THEN 2070
2110 D$ = "PLONS"
2120 IF A(R,K) > 6 THEN 2190
2130 X = A(R,K)
2140 A(R,K) = X + 6
2150 IF X = 1 THEN 2200
2160 B(X-1,3) = B(X-1,3) - 1
2170 D$ = B$(X - 1)
2180 REM ** ZEE-AFDRUK **
```

```
2190 D$ = "-RESULTAAT- " + D$
2200 GOSUB 1300
2210 H = 0
2220 FOR I = 1 TO 5
2230 IF B(I,3) <> 0 THEN 2250
2240 H = H + 1
2250 NEXT I
2260 IF H <> 5 THEN 2280
2270 T = 100                :REM SPELER GEWONNEN
2280 RETURN
2290 :
2300 REM *** ONBEKEND H = 1 ***
2310 REM ** KOLOM ? **
2320 A1 = INT(RND(0) * 10 + 1)
2330 A2 = 10
2340 B1 = 1
2350 B2 = 10
2360 L = INT(RND(0) * 2)
2370 H = -1
2380 IF L = 0 THEN 2490
2390 REM ** REGEL ? **
2400 B1 = A1
2410 A1 = 1
2420 IF H = -1 THEN 2490
2430 A2 = A1
2440 A1 = 1
2450 IF L = 0 THEN 2490
2460 B2 = B1
2470 B1 = 1
2480 A2 = 10
2490 H = H + 1
2500 FOR I = A1 TO A2
2510 FOR J = B1 TO B2
2520 IF A(I+Q*10,J) < 0 OR A(I+Q*10,J) > 6 THEN 2580
2530 R = I
2540 K = J
2550 H = 9
2560 I = A2
2570 J = B2
2580 NEXT J
2590 NEXT I
2600 IF H = 0 THEN 2430
2610 RETURN
2620 :
2630 REM *** SCHOT COMPUTER ***
2640 Q = 1
2650 S = 1
2660 H = 1
2670 RESTORE
2680 FOR I = 1 TO 5
2690 READ Y,V
2700 IF B(I,2) < 0 OR B(I,1) = 0 THEN 2760
2710 X = I
2720 H = 2
2730 IF B(I,2) = 1 THEN 2750
```

```
2740 H = 3
2750 I = 5
2760 NEXT I
2770 ON H GOSUB 2320 ,3130 ,2850
2780 S = 1
2790 D$ = ""
2800 GOSUB 1590
2810 RETURN
2820 :
2830 REM *** BESCHADIGD H = 3 ***
2840 REM * VERTIKAAL *
2850 L = 0                    :REM HULPVAR.
2860 H = 0                    :REM HULPVAR
2870 R = INT(B(X,1)/100)
2880 K = B(X,1)-R*100
2890 FOR I = R - Y TO R + Y
2900 IF I < 0 OR I > 10 THEN 2940
2910 IF A(I+Q*10,K) <> V + 6 OR I = R THEN 2940
2920 H = 1
2930 I = R + Y
2940 NEXT I
2950 IF H = 1 THEN 2980  :REM VERTICAAL
2960 REM * HORIZONTAAL *
2970 L = 1
2980 X = 1
2990 I = R
3000 J = K
3010 X = X * -1
3020 I = I + (H * X)
3030 J = J + (L * X)
3040 IF I < 1 OR J < 1 THEN 3010
3050 IF I > 10 OR J > 10 THEN 3100
3060 IF A(I+Q*10,J) = V + 6 THEN 3020
3070 IF A(I+Q*10,J) = -1 OR A(I+Q*10,J) = 7 THEN 3010
3080 R = I
3090 K = J
3100 RETURN
3110 :
3120 REM *** AANGESCHOTEN H = 2 ***
3130 R1 = INT(B(X,1)/100)
3140 K1 = B(X,1) - R1 * 100
3150 FOR I = R1 - 1 TO R1 + 1
3160 FOR J = K1 - 1 TO K1 + 1
3170 IF I < 1 OR J < 1 THEN 3260
3180 IF I > 10 OR J > 10 THEN 3260
3190 IF I <> R1 AND J <> K1 THEN 3260
3200 IF I = R1 AND J = K1 THEN 3260
3210 IF A(I+Q*10,J) = -1 OR A(I+Q*10,J) = 7 THEN 3260
3220 R = I
3230 K = J
3240 I = R1 + 1
3250 J = K1 + 1
3260 NEXT J
3270 NEXT I
3280 RETURN
```

```
3290 :
3300 REM *** RESULTAAT ***
3310 D$ = "SPELER "
3320 Q = 0
3330 S = 1
3340 IF T = 100 THEN 3380
3350 Q = 1
3360 S = 0
3370 D$ = "COMPUTER "
3380 D$ = D$ + "GEWONNEN"
3390 GOSUB 1300
3400 RETURN
3410 :
3420 REM **** STUURMODULE ****
3430 REM *** INITIALISEREN ***
3440 GOSUB 120
3450 REM *** SPEELVELD OPBOUWEN COMPUTER ***
3460 GOSUB 360
3470 REM *** SPEELVELD SPELER OPBOUWEN ***
3480 GOSUB 920
3490 REM *** SCHOT SPELER ***
3500 GOSUB 2010
3510 IF T = 100 THEN 3560
3520 REM *** SCHOT COMPUTER ***
3530 GOSUB 2640
3540 IF T < 5 THEN 3500
3550 REM *** RESULTAAT ***
3560 GOSUB 3310
3570 END
```

Voorbeeld

```
*** ZEESLAG ***
                    - COMPUTER -
       1  2  3  4  5  6  7  8  9  10
 1     .  .  .  .  .  .  .  .  .  .
 2     .  .  .  .  .  .  .  .  .  .
 3     .  .  .  .  .  .  .  .  .  .
 4     .  .  .  .  .  .  .  .  .  .
 5     .  .  .  .  .  .  .  .  .  .
 6     .  .  .  .  .  .  .  .  .  .
 7     .  .  .  .  .  .  .  .  .  .
 8     .  .  .  .  .  .  .  .  .  .
 9     .  .  .  .  .  .  .  .  .  .
 10    .  .  .  .  .  .  .  .  .  .
GEEF X , Y - COORD  5 , 5

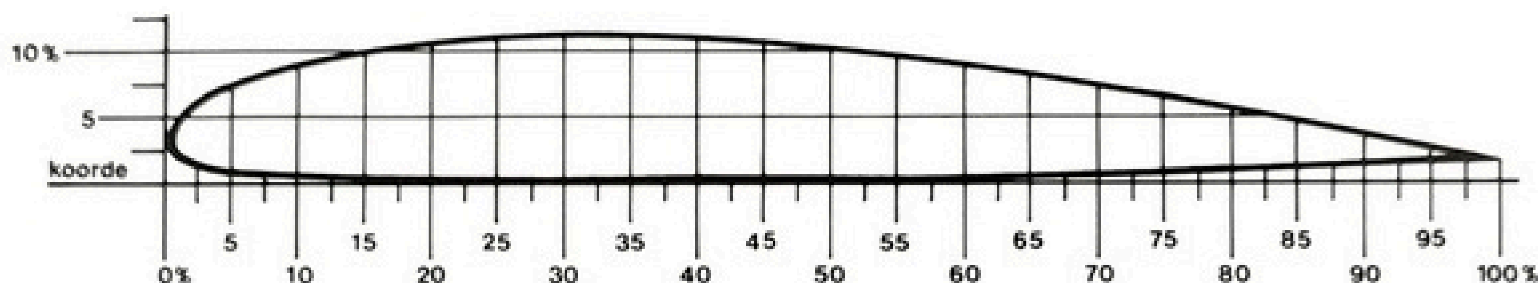

*** ZEESLAG ***    PLONS
                    - COMPUTER -
                       *
       1  2  3  4  5  6  7  8  9  10
 1     .  .  .  .  .  .  .  .  .  .
 2     .  .  .  .  .  .  .  .  .  .
 3     .  .  .  .  .  .  .  .  .  .
 4     .  .  .  .  .  .  .  .  .  .
 5 *   .  .  .  .  *  .  .  .  .  .
 6     .  .  .  .  .  .  .  .  .  .
 7     .  .  .  .  .  .  .  .  .  .
 8     .  .  .  .  .  .  .  .  .  .
 9     .  .  .  .  .  .  .  .  .  .
 10    .  .  .  .  .  .  .  .  .  .


*** ZEESLAG ***
                    - SPELER -
                          *
       1  2  3  4  5  6  7  8  9  10
 1 *  -V- .  .  .  .  . -M- .  . -K-
 2    -V- .  .  .  .  . -M- .  . -K-
 3    -V- .  .  .  .  . -M- .  . -K-
 4    -V- .  .  .  .  .  .  .  . -K-
 5    -V- .  .  .  .  .  .  .  .  .
 6     .  .  .  .  .  .  .  .  .  .
 7     .  .  .  .  .  .  .  .  .  .
 8     .  .  .  .  .  .  .  .  .  .
 9     .  .  .  .  .  .  .  .  .  .
 10   -F--F--F- .  .  .  .  . -D--D-
IS ER IETS GERAAKT <JA/NEE> NEE
```

4. Algemene programma's

4.1 Vleugelprofiel

Een van de vervelendste klusjes in de vliegtuigmodelbouw is het uitzetten van profielen aan de hand van profielgegevens. De keuze van het profiel is vaak een moeilijke zaak omdat het profiel de vluchtkarakteristiek in sterke mate beïnvloedt. Een zweefvliegtuig met een bol-bolprofiel lijkt dan ook ondenkbaar, maar een radiobestuurd modelvliegtuig zou best uitgerust kunnen zijn met zo'n bol-bolprofiel (bijvoorbeeld een NACA 0018 profiel).



Vaak vindt men de bekende profielen in modelbouwtijdschriften of boeken over aerodynamica. In een aantal bibliotheken zijn hele dikke boeken met profielgegevens.
De maatgegevens van een profiel zijn gestandaardiseerd. Zo zijn alle gegevens van het profiel uitgedrukt in een percentage van de koorde. De koorde wordt op 100% gesteld. De maten van de profielen worden in drie kolommen weergegeven. In de eerste kolom wordt de afstand aangegeven vanaf de neus van het profiel. In de tweede kolom wordt de bovenzijde van het profiel gegeven t.o.v. de hartlijn. De derde kolom geeft de onderzijde van het profiel.
Als voorbeeld om het bovenstaande te verduidelijken is gekozen voor het bekende Clark YH profiel. Dit profiel wordt veel gebruikt voor eenvoudige radiobestuurde vliegtuigen.

CLARK YH

afstand gemeten vanaf de neus van het profiel	bovenzijde	onderzijde
0.0	3.50	3.50
1.25	5.45	1.93
2.50	6.50	1.47
5.00	7.90	0.93
7.50	8.85	0.63
10.00	9.60	0.42
15.00	10.68	0.15
20.00	11.36	0.03
30.00	11.70	0.0
40.00	11.40	0.0
50.00	10.51	0.0
60.00	9.15	0.0
70.00	7.42	0.06
80.00	5.62	0.38
90.00	3.84	1.02
95.00	2.93	1.40
100.00	2.05	1.85

Nadat alle gegevens zijn ingevoerd, kunnen nog correcties worden aangebracht. Het is niet denkbeeldig dat er fouten zijn gemaakt bij de invoer. Om te kunnen corrigeren moet men het regelnummer geven. Wil men de gegevens op de regel alleen bekijken en niet wijzigen, dan tikt men bij de eerste vraag −9 in. Heeft men helemaal niets meer te wijzigen, dan geeft men bij 'WELKE REGEL WILT U WIJZIGEN' een nul.
Ten slotte geeft men op, welke koorde het profiel moet krijgen. Deze afstand van neus tot aan achterligger wordt opgegeven in millimeters. Heeft men alle gegevens ingevoerd dan vindt de berekening plaats. De berekening kan herhaald worden met dezelfde profielgegevens, maar met een andere koorde.
Wanneer men het programma wil verlaten, geeft men bij de vraag over de koorde van het profiel een nul.
Heeft men een computer of printer, waarop men 'high resolution' plaatjes kan maken, dan zou men kunnen overwegen het profiel op het beeldscherm of printer uit te zetten.

Programma

```
10 REM *** VLEUGELPROFIEL ***
20 DIM A(25,3), B(25,3)
30 GOTO 1360
40 :
50 REM *** INITITIALISEREN ***
60 FOR I = 1 TO 25
70 FOR J = 1 TO 3
80 A(I,J) = 0
90 NEXT J
100 NEXT I
110 T = 0
120 RETURN
130 :
140 REM *** INVOER ***
150 FOR I = 1 TO 25
160 IF (I-1)/3 - INT((I-1)/3) <> 0 THEN 210
170 REM ** KOP AFDRUKKEN **
180 GOSUB 1020
190 PRINT
200 REM ** INPUT-BLOK **
210 X = I
220 GOSUB 1080
230 IF H <> 0 THEN 260
240 T = I
250 I = 25             :REM LUS-TELLER OP MAXIMUM
260 NEXT I
270 IF T <> 0 THEN 290
280 T = 25
290 RETURN
300 :
310 REM *** LENGTE PROFIEL ***
320 REM ** KOP AFDRUKKEN **
330 GOSUB 1020
340 PRINT "GEEF DE LENGTE IN MILLIMETERS WAAROP"
350 PRINT "HET PROFIEL MOET WORDEN UITGEZET."
360 PRINT "<STOP = 0>   HOEVEEL MILLIMETER ";
370 INPUT M
380 IF M < 0 THEN 340
390 RETURN
400 :
410 REM *** CORRECTIE-ROUTINE ***
420 REM ** KOP AFDRUKKEN **
430 GOSUB 1020
440 PRINT "WELKE REGEL WIJZIGEN  <STOP = 0> ";
450 INPUT X
460 IF X = 0 THEN 570
470 IF X < 0 OR X > T THEN 440
480 PRINT "OPGENOMEN GEGEVENS"
490 PRINT "REGEL ......."; X
500 PRINT "% VANAF NEUS ";A(X,1)
510 PRINT "% BOVENZIJDE ";A(X,2)
520 PRINT "% ONDERZIJDE ";A(X,3)
530 PRINT "INDIEN U NIET WILT WIJZIGEN TOETS BIJ DE"
```

```
540 PRINT "EERSTE VRAAG -9 + <RETURN> "
550 REM ** INPUT-BLOK **
560 GOSUB 1080
570 RETURN
580 :
590 REM *** BEWERKING ***
600 FOR I = 1 TO T
610 FOR J = 1 TO 3
620 B(I,J ) = M * A(I,J)/100               :REM MAAT BEPALEN
630 B(I,J) = INT(B(I,J) * 10 + 0.5)/10     :REM AFRONDEN
640 NEXT J
650 NEXT I
660 RETURN
670 :
680 REM *** UITVOER ***
690 FOR I = 1 TO T
700 IF (I-1)/5 - INT((I-1)/5) <> 0 THEN 810
710 IF I = 1 THEN 760
720 REM ** WACHTEN **
730 GOSUB 1250
740 IF H = 0 THEN 990
750 REM ** KOP AFDRUKKEN **
760 GOSUB 1020
770 PRINT "PROFIEL UITGEZET OP ";M;"MM."
780 PRINT "MATEN IN MILLIMETERS !"
790 PRINT
800 PRINT "VANAF NEUS  BOVENZIJDE  ONDERZIJDE"
810 FOR J = 1 TO 3
820 A$ = "0.0"
830 IF B(I,J) = 0 THEN 920
840 B$ = "0"
850 IF INT(B(I,J)) < 0.001 THEN 870
860 B$ = ""
870 A$ = STR$(B(I,J))             :REM NUMERIEK => ALFA
880 A$ = RIGHT$(A$,LEN(A$)-1)
890 C$ = RIGHT$(A$,2)
900 IF LEFT$(C$,1) = "." THEN 920
910 A$ =  A$ + ".0"
920 A$ = B$ + A$
930 L = 10 - LEN(A$)
940 PRINT TAB((J-1) * 12 +  L)A$;
950 NEXT J
960 PRINT
970 NEXT I
980 GOSUB 1250          :REM ** WACHTEN **
990 RETURN
1000 :
1010 REM *** KOP AFDRUKKEN ***
1020 CLS                   :REM SCHERM WISSEN
1030 PRINT "*** VLEUGELPROFIEL ***"
1040 PRINT
1050 RETURN
1060 :
```

```
1070 REM *** INPUT BLOK ***
1080 H = 0
1090 PRINT X;"GEEF % VANAF NEUS PROFIEL ";
1100 INPUT A
1110 IF A = -9 AND T <> 0 THEN 1220
1120 IF A > 100 THEN 1090
1130 A(X,1) = A
1140 IF A = 100 THEN 1160
1150 H = 1
1160 PRINT X;"GEEF % BOVENZIJDE ";
1170 INPUT A(X,2)
1180 PRINT X;"GEEF % ONDERZIJDE ";
1190 INPUT A(X,3)
1200 IF A(X,2) < A(X,3) THEN 1160
1210 PRINT
1220 RETURN
1230 :
1240 REM *** WACHTEN ***
1250 PRINT "WILT U DOORGAAN <JA/NEE> ";
1260 INPUT A$
1270 A$ = LEFT$(A$,1)
1280 H = 0                    :REM STUURVARIABELE
1290 IF A$ = "N" THEN 1320
1300 H = 1
1310 IF A$ <> "J" THEN 1250
1320 RETURN
1330 :
1340 REM **** STUURMODULE ****
1350 REM *** INITIALISEREN ***
1360 GOSUB 60
1370 REM *** INVOER ***
1380 GOSUB 150
1390 REM *** CORRECTIE ***
1400 GOSUB 430
1410 IF X <> 0 THEN 1400
1420 REM *** LENGTE PROFIEL ***
1430 GOSUB 330
1440 IF M = 0 THEN 1500
1450 REM *** BEREKENING ***
1460 GOSUB 600
1470 REM *** UITVOER ***
1480 GOSUB 690
1490 IF H = 1 THEN 1430
1500 END
```

Voorbeeld

```
*** VLEUGELPROFIEL ***

 1 GEEF % VANAF NEUS PROFIEL  0
 1 GEEF % BOVENZIJDE  3.5
 1 GEEF % ONDERZIJDE  3.5

 2 GEEF % VANAF NEUS PROFIEL  1.25
 2 GEEF % BOVENZIJDE  5.45
 2 GEEF % ONDERZIJDE  1.93

 3 GEEF % VANAF NEUS PROFIEL  2.5
 3 GEEF % BOVENZIJDE  6.5
 3 GEEF % ONDERZIJDE  1.47

*** VLEUGELPROFIEL ***

 4 GEEF % VANAF NEUS PROFIEL  5
 4 GEEF % BOVENZIJDE  7.9
 4 GEEF % ONDERZIJDE  .93

 5 GEEF % VANAF NEUS PROFIEL  7.5
 5 GEEF % BOVENZIJDE  8.85
 5 GEEF % ONDERZIJDE  .63

 6 GEEF % VANAF NEUS PROFIEL  10
 6 GEEF % BOVENZIJDE  9.6
 6 GEEF % ONDERZIJDE  .42

*** VLEUGELPROFIEL ***

 16 GEEF % VANAF NEUS PROFIEL  95
 16 GEEF % BOVENZIJDE  2.93
 16 GEEF % ONDERZIJDE  1.4

 17 GEEF % VANAF NEUS PROFIEL  100
 17 GEEF % BOVENZIJDE  2.05
 17 GEEF % ONDERZIJDE  1.85

*** VLEUGELPROFIEL ***

WELKE REGEL WIJZIGEN  <STOP = 0>  0

*** VLEUGELPROFIEL ***

GEEF DE LENGTE IN MILLIMETERS WAAROP
HET PROFIEL MOET WORDEN UITGEZET.
<STOP = 0>   HOEVEEL MILLIMETER  1000
```

```
*** VLEUGELPROFIEL ***

PROFIEL UITGEZET OP  1000 MM.
MATEN IN MILLIMETERS !

VANAF NEUS  BOVENZIJDE  ONDERZIJDE
       0.0        35.0        35.0
      12.5        54.5        19.3
      25.0        65.0        14.7
      50.0        79.0         9.3
      75.0        88.5         6.3
WILT U DOORGAAN <JA/NEE> JA
```

```
*** VLEUGELPROFIEL ***

PROFIEL UITGEZET OP  1000 MM.
MATEN IN MILLIMETERS !

VANAF NEUS  BOVENZIJDE  ONDERZIJDE
     100.0        96.0         4.2
     150.0       106.8         1.5
     200.0       113.6         0.3
     300.0       117.0         0.0
     400.0       114.0         0.0
WILT U DOORGAAN <JA/NEE>
```

4.2 Kalender

Dit programma maakt voor een bepaald jaar de kalender aan met de aanduidingen van de dagen. Hiertoe geeft men bij de vraag 'welk jaar' het jaar in vier cijfers en bij de vraag 'welke maand' het maandnummer dat men hebben wil. Wil men alle maanden van het jaar hebben dan geeft men bij de vraag 'welke maand' een nul.
Het programma is gebaseerd op de berekening van Zeller.



Dit programma is ook als subroutine in andere programma's te gebruiken. Bijvoorbeeld in een programma, verjaardagskalender. In een interne data-file plaatst men alle namen van familie, vrienden en kennissen van wie men de verjaardag wil onthouden met de bijbehorende verjaardagsdatum. Wordt een maand op het beeldscherm gezet, dan worden tegelijkertijd alle verjaardagen aangegeven van mensen die in de desbetreffende maand jarig zijn.

Programma

```
10 REM *** KALENDER ***
20 DIM A(7,6)
30 GOTO 1070
40 :
50 REM *** INITIALISEREN ***
60 C$ = "303232332323"
70 D$ = "MADIWODOVRZAZO"
80 G$ = "JANFEBMRTAPRMEIJUNJULAUGSEPOKTNOVDEC "
90 RETURN
100 :
110 REM *** MAANDBEREKENING ***
120 REM ** BEPALING LENGTE VAN DE MAAND **
130 E$ = LEFT$(C$,M)
140 D = 28 + VAL(RIGHT$(E$,1))
150 REM ** BEPALING SCHRIKKELJAAR **
160 IF Y / 4 - INT(Y/4) <> 0 THEN 200
170 IF Y = 2000 OR M <> 2 THEN 200
180 D = D + 1
190 REM *** BEPALING EERSTE V.D. MAAND ***
200 GOSUB 910
210 REM *** MAT A OPBOUWEN ***
220 H = 1
230 FOR J = 1 TO 6
240 FOR I = 1 TO 7
250 P = I + (J - 1) * 7
260 A(I,J) = 0
270 IF P < A OR H > D THEN 300
280 A(I,J) = H
290 H = H + 1
300 NEXT I
310 NEXT J
320 REM ** BEPALING NAAM VAN DE MAAND **
330 H$ = LEFT$(G$, M * 3)
340 H$ = RIGHT$(H$,3)
350 RETURN
360 :
370 REM *** UITVOER ***
380 CLS                     :REM SCHERM WISSEN
390 PRINT "*** KALENDER ***"
400 PRINT "MAAND ";H$;Y
410 PRINT
420 FOR I = 1 TO 7
430 E$ = LEFT$(D$, 2 * I)
440 PRINT RIGHT$(E$,2);
450 FOR J = 1 TO 6
460 IF A(I,J) = 0 THEN 480
470 PRINT TAB(J * 4 + 3) A(I,J);
480 NEXT J
490 PRINT
500 NEXT I
510 RETURN
520 :
530 REM *** INVOER ***
```

```
540 CLS                         :REM SCHERM WISSEN
550 PRINT "*** KALENDER ***"
560 PRINT
570 PRINT "WELK JAAR -4 CIJFERS- <STOP = 0> ";
580 INPUT Y
590 IF Y = 0 THEN 640
600 IF Y < 1700 OR Y > 2200 THEN 570
610 PRINT "WELKE MAAND    <ALLE = 0> ";
620 INPUT K
630 IF K < 0 OR K > 12 THEN 610
640 RETURN
650 :
660 REM *** BESTURING ***
670 FOR Z = 1 TO 12
680 M = Z
690 IF K = 0 THEN 710
700 M = K
710 GOSUB 110
720 REM ** UITVOER **
730 GOSUB 380
740 REM ** INVOER II **
750 GOSUB 820
760 IF K = 0 THEN 780
770 Z = 12                      :REM UIT LUS
780 NEXT Z
790 RETURN
800 :
810 REM *** INVOER II ***
820 PRINT "WILT U DOORGAAN  <JA/NEE> ";
830 INPUT L$
840 L$ = LEFT$(L$,1)
850 IF L$ = "J" THEN 880
860 IF L$ <> "N" THEN 820
870 K = 1
880 RETURN
890 :
900 REM *** BEPALING EERSTE V.D. MAAND ***
910 P =M
920 L = Y
930 IF P > 2 THEN 960
940 P = P + 12
950 L = L - 1
960 E = INT(L/100)
970 R = L - E * 100
980 A = INT(2.6001 * (P-2) - 0.2)
990 A = A + 1 + R + INT(R/4) + INT(E/4) - 2 * E
1000 A = A - INT(A/7) * 7
1010 IF A <> 0 THEN 1030
1020 A = 7
1030 RETURN
1040 :
1050 REM **** STUURMODULE ****
1060 REM *** INITIALISEREN ***
1070 GOSUB 60
1080 REM *** INVOER ***
```

```
1090 GOSUB 540
1100 IF Y = 0 THEN 1140
1110 REM *** BESTURING ***
1120 GOSUB 670
1130 IF L$ = "J" THEN 1090
1140 END
```

Voorbeeld

```
*** KALENDER ***

WELK JAAR -4 CIJFERS- <STOP = 0>  1983
WELKE MAAND   <ALLE = 0>  6


*** KALENDER ***
MAAND JUN 1983

MA          6   13  20  27
DI          7   14  21  28
WO      1   8   15  22  29
DO      2   9   16  23  30
VR      3   10  17  24
ZA      4   11  18  25
ZO      5   12  19  26
WILT U DOORGAAN  <JA/NEE> NEE
```

4.3 Chi-kwadraat

De chi-kwadraat toets wordt onder andere gebruikt om na te gaan of bepaalde uitkomsten kloppen. Met een voorbeeld is dit snel duidelijk te maken. Stel dat men een dobbelsteen heeft waarvan men zegt dat het een zuivere dobbelsteen is. Volgens de theorie moet gelden dat elk vlak van de dobbelsteen evenveel kans heeft om voor te komen. Zou men met zo'n dobbelsteen 600 keer gooien dan komt volgens de theorie de een honderd keer voor, de twee honderd keer, de drie honderd keer, de vier honderd keer, de vijf honderd keer en de zes honderd keer. Wanneer men ten slotte het experiment in werkelijkheid uitvoert, komen de bovengenoemde getallen er niet uit, maar de cijfers volgens onderstaand staatje.

aantal ogen	aantal worpen
1	102
2	98
3	97
4	98
5	103
6	102
totaal	600

Nu is het de vraag of de dobbelsteen met 95% zekerheid een zuivere dobbelsteen is. Voert men de gegevens in, in het programma dan kan men het antwoord vinden.



Andere voorbeelden waar de Chi-kwadraat toets bij gebruikt wordt, is o.a. bij de controle op vulmachines.

Programma

```
10 REM *** CHI-KWADRAAT ***
20 DIM A(25,2)
30 DATA 3.841,5.991,7.815,9.488,11.07
40 DATA 12.59,14.07,15.51,16.92,18.31
50 DATA 19.68,21.03,22.36,23.68,25.00
60 DATA 26.30,27.59,28.87,30.14,31.41
70 DATA 32.67,33.92,35.17,36.42,37.65
80 DATA 38.88,40.11,41.34,42.56,43.77
90 GOTO 790
100 :
110 REM *** INITIALISEREN ***
120 T = 0
130 X = 0                    :REM CHI-KWADRAAT
140 REM ** MAT A SCHOONVEGEN **
150 FOR I = 1 TO 25
160 A(I,1) = 0
170 A(I,2) = 0
180 NEXT I
190 RETURN
200 :
210 REM *** INVOER ***
220 FOR I = 1 TO 25
230 IF (I - 1) / 5 - INT((I-1)/5) <> 0 THEN 280
240 REM ** KOP AFDRUKKEN **
250 GOSUB 730
260 PRINT "INDIEN GEEN INVOER,TOETS STOP + <RETURN>"
270 PRINT
280 PRINT "GEEF THEORETISCHE WAARNEMING ";
290 INPUT A$
300 H = 0
310 IF LEFT$(A$,1) = "S" THEN 380
320 H = 1
330 PRINT "GEEF DE FEITELIJKE WAARNEMING ";
340 INPUT B$
350 A(I,1) = VAL(A$)
360 A(I,2) = VAL(B$)
370 IF H = 1 THEN 400
380 T = I - 1
390 I = 25                   :REM LUS-TELLER OP MAXIMUM
400 NEXT I
410 IF H = 0 THEN 430
420 T = 25
430 RETURN
440 :
450 REM *** BEREKENING ***
460 FOR I = 1 TO T
470 X = X + (A(I,1) - A(I,2)) ↑ 2 / A(I,2)
480 NEXT I
490 V = T - 1                   :REM AANTAL VRIJHEIDSGRADEN
```

```
500 RESTORE                          :REM DATAPOINTER VOORAAN
510 FOR I = 1 TO V
520 READ K
530 NEXT I
540 H = 0
550 IF X < K THEN 570
560 H = 1
570 RETURN
580 :
590 REM *** UITVOER ***
600 REM ** KOP AFDRUKKEN **
610 GOSUB 730
620 PRINT
630 PRINT "CHI-KWADRAAT  :";X
640 PRINT "AANTAL VRIJHEIDSGRADEN ";V
650 PRINT "KRITIEKE WAARDE BIJ 5 % IS ";K
660 A$ = "H0 WORDT GEACCEPTEERD."
670 IF H = 0 THEN 690
680 A$ = "H0 WORDT VERWORPEN."
690 PRINT A$
700 RETURN
710 :
720 REM *** KOP AFDRUKKEN ***
730 CLS                            :REM SCHERM WISSEN
740 PRINT "*** CHI-KWADRAAT ***"
750 RETURN
760 :
770 REM **** STUURMODULE ****
780 REM *** INITIALISEREN ***
790 GOSUB 120
800 REM *** INVOER ***
810 GOSUB 220
820 REM *** BEREKENING ***
830 GOSUB 460
840 REM *** UITVOER ***
850 GOSUB 610
860 END
```

Voorbeeld

```
*** CHI-KWADRAAT ***
INDIEN GEEN INVOER,TOETS STOP + <RETURN>

GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 102
GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 98
GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 97
GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 98
GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 103


*** CHI-KWADRAAT ***
INDIEN GEEN INVOER,TOETS STOP + <RETURN>

GEEF THEORETISCHE WAARNEMING 100
GEEF DE FEITELIJKE WAARNEMING 102
GEEF THEORETISCHE WAARNEMING STOP


*** CHI-KWADRAAT ***

CHI-KWADRAAT  : .340226
AANTAL VRIJHEIDSGRADEN  5
KRITIEKE WAARDE BIJ 5 % IS  11.07
H0 WORDT GEACCEPTEERD.
```

4.4 Huishoudboekje

Hoe vaak wordt er niet gezegd 'waar blijft mijn geld? Waaraan heb ik het uitgegeven?' Een van de eerste oplossingen is: opschrijven! Maar dan komt het. Hoe vergelijk je uitgaven met elkaar? Hoeveel procent geven we aan kleding uit? Wat hebben we in januari aan eten en drinken uitgegeven en wat hebben we er tot nu toe aan uitgegeven? Dit soort vragen zijn met dit programma op een eenvoudige manier te beantwoorden zonder dat men zelf hoeft te rekenen. Het enige dat men moet doen, is zorgen dat de interne data-file met de gegevens gevuld staat en de vragen goed worden beantwoord.
Om uitgaven in de interne data-file op te nemen, moeten zij op een bepaalde manier gecodeerd staan. Men begint met een rekeningnummer. De codes komen overeen met de indexen op regel 60 t/m 140. Daarna volgt de datum op een genormaliseerde manier geschreven. Dit houdt in dat men op de eerste twee plaatsen van het getal het jaar aangeeft, de volgende twee plaatsen zijn voor de maand en de laatste twee zijn voor de dag. Bijvoorbeeld 14 februari 1983 wordt dan 830214. Na het rekeningnummer en de datum volgt tussen aanhalingstekens de omschrijving van de aankoop en ten slotte het bedrag. Alle onderdelen worden in de DATA-regel gescheiden door komma's. Men hoeft bij invoer niet te letten op de data, deze komen automatisch in de goede maand terecht. Wel dient men erop te letten dat de laatste DATA-regel 999 bevat (zie programmaregel 2260). Deze 999 geeft aan in het programma dat het einde van de interne data-file bereikt is.
In het programma bestaan vier verschillende keuzemogelijkheden. Men kan de totalen per rekening krijgen tot en met de geboekte datum of over een specifieke maand. Men kan ook een specifieke rekening krijgen tot en met de laatste boekingsdatum of een bepaalde rekening over een speciale maand.

Bij verschillende computers moet men er op letten dat de nauwkeurigheid van de getallen geen geweld wordt aangedaan. In de meeste computers geldt voor numerieke floating point-variabelen een nauwkeurigheid van acht tot negen cijfers. In dat geval zullen er niet snel problemen ontstaan. Is de nauwkeurigheid beperkt tot zes cijfers dan kunnen er vrij snel problemen ontstaan. Een oplossing is dan de floating point-variabelen, die de bedragen manipuleren, als dubbel precision getallen te definiëren.

Programma

```
10 REM *** HUISHOUDBOEKJE ***
20 DIM A(9),B$(9)
30 GOTO 1960
40 :
50 REM *** INITIALISERN ***
60 B$(1) = "LEVENSMIDDELEN"
70 B$(2) = "VAKANTIE / UITGAAN"
80 B$(3) = "AUTO/MOTOR/FIETS"
90 B$(4) = "VERZEKERINGEN"
100 B$(5) = "ENERGIE EN WATER"
110 B$(6) = "WOONLASTEN"
120 B$(7) = "KLEDING"
130 B$(8) = "STUDIE EN OVERIGEN"
140 B$(9) = "OVERIGEN"
150 RETURN
160 :
170 REM *** INVOER ***
180 CLS                       :REM SCHERM WISSEN
190 PRINT "*** HUISHOUDBOEKJE *** "
200 PRINT
210 PRINT "0 = STOPPEN"
220 PRINT "1 = REKENING OVER EEN BEPAALDE MAAND"
230 PRINT "2 = REKENING T/M GEBOEKTE DATUM    "
240 PRINT "3 = TOTALEN OVER EEN BEPAALDE MAAND"
250 PRINT "4 = TOTALEN T/M GEBOEKTE DATUM"
260 PRINT
270 PRINT "WAT IS UW KEUZE ";
280 INPUT K
290 IF K = 0 THEN 370
300 IF K < 0 OR K > 4 THEN 180
310 IF K > 2 THEN 340
320 REM ** INVOER REKENING-NUMMER **
330 GOSUB 400
340 IF K = 2 OR K = 4 THEN 370
350 REM ** INVOER MAAND **
360 GOSUB 530
370 RETURN
380 :
390 REM *** INVOER REKENINGNUMMER ***
400 CLS                       :REM SCHERM WISSEN
410 PRINT "*** HUISHOUDBOEKJE ***"
420 PRINT
430 PRINT TAB(4)"0 =  ALLE REKENINGEN"
440 FOR I = 1 TO 9
450 PRINT TAB(3)I;TAB(5)"=  ";B$(I)
460 NEXT I
470 PRINT "WELK REKENINGNUMMER KIEST U ";
480 INPUT R
490 IF R < 0 OR R > 9 THEN 470
500 RETURN
510 :
520 REM *** INVOER MAAND ***
530 PRINT "WELKE MAAND KIEST U ";
```

```
540 INPUT M
550 IF M < 1 OR M > 12 THEN 530
560 RETURN
570 :
580 REM *** INLEZEN RECORD UIT INTERNE DATA FILE ***
590 H = 0
600 READ N                  :REM REKENINGNUMMER
610 IF N = 999 THEN 730
620 H = 1                   :REM NOG GEEN EOF
630 READ D,A$,B             :REM DATUM,OMSCHRIJVING,BEDRAG
640 IF N < 1 OR N > 9 THEN 590
650 IF K = 4 THEN 730
660 IF K = 2 THEN 720
670 REM ** BEPAALDE MAAND **
680 J = INT(D / 10000)
690 Z = INT((D - J * 10000) / 100)
700 IF Z <> M THEN 590
710 IF K = 3 THEN 730
720 IF G <> N THEN 590
730 RETURN
740 :
750 REM *** VOORBEWERKING ***
760 REM ** SCHOONVEGEN MAT A **
770 FOR I = 1 TO 9
780 A(I) = 0
790 NEXT I
800 REM ** VARIABELEN OP STARTWAARDE ZETTEN **
810 S = 0                   :REM TOTAAL GENERAAL
820 M = 0                   :REM MAAND ALLES
830 R = 0                   :REM REKENING ALLES
840 P = 0                   :REM PRINTREGELTELLER
850 RESTORE                 :REM DATAPOINTER VOORAAN
860 RETURN
870 :
880 REM *** BEWERKING - TOTALEN ***
890 REM ** RECORD LEZEN **
900 GOSUB 590
910 IF H = 0 THEN 950
920 A(N) = A(N) + B                :REM TOTALISEREN
930 S = S + B
940 IF H = 1 THEN 900
950 RETURN
960 :
970 REM *** UITVOER - TOTALEN ***
980 CLS                    :REM SCHERM WISSEN
990 PRINT "*** HUISHOUDBOEKJE *** "
1000 A$ = "T/M GEBOEKTE MAAND "
1010 IF K = 4 THEN 1030
1020 A$ = "OVER MAAND : "+STR$(M)
1030 PRINT A$
1040 PRINT
1050 FOR I = 1 TO 9
1060 B = A(I)
1070 REM ** PERCENTAGE'S **
1080 E = 0
```

```
1090 IF S = 0 THEN 1120
1100 E = INT(B / S * 1000 + 0.5)/10
1110 REM ** POSITIONEREN BEDRAG **
1120 GOSUB 1240
1130 PRINT I;TAB(3) B$(I);TAB(22+L)A$;
1140 PRINT TAB(33)E;TAB(38)"%"
1150 NEXT I
1160 PRINT TAB(20)"---------------"
1170 IF S = 0 THEN 1210
1180 B = S
1190 GOSUB 1240
1200 PRINT TAB(3) "TOTAAL";TAB(22 + L)A$;TAB(34)"100%"
1210 RETURN
1220 :
1230 REM *** POSITIONEREN BEDRAG ***
1240 B$ = "--"
1250 W = B - INT(B)          :REM GETAL ACHTER KOMMA
1260 A$ = STR$(INT(B))
1270 IF W = 0 THEN 1320
1280 W = INT(W * 100 + 0.5)
1290 B$ = STR$(W)
1300 B$ = "00" + RIGHT$(B$,LEN(B$)-1)
1310 B$ = RIGHT$(B$,2)
1320 A$ = A$ + "," + B$
1330 IF B <> 0 THEN 1350
1340 A$ = "-    "
1350 L = 10 - LEN(A$)
1360 RETURN
1370 :
1380 REM *** BEWERKING-REKENING ***
1390 FOR Q = 1 TO 9
1400 RESTORE                          :REM DATAPOINTER VOORAAN
1410 S = 0
1420 P = 0                  :REM BLADTELLER OP STARTWAARDE
1430 G = Q
1440 IF R = 0 THEN 1470
1450 G = R
1460 REM ** RECORD LEZEN **
1470 GOSUB 590
1480 IF H = 0 THEN 1550
1490 A(G) = A(G) + B
1500 S = S + B
1510 REM ** REGEL UITVOER **
1520 GOSUB 1660
1530 IF H = 1 THEN 1470
1540 REM ** TOTAALREGEL OPBOUWEN **
1550 D = 0
1560 A$ = "TOTAAL REKENING"
1570 B = S
1580 P = -9
1590 GOSUB 1660
1600 IF R = 0 AND A$ = "J" THEN 1620
1610 Q = 10                 :REM LUS-TELLER MAXIMAAL
1620 NEXT Q
1630 RETURN
```

```
1640 :
1650 REM *** REGEL AFDRUKKEN ***
1660 IF P <> 0 AND S <> 0 THEN 1720
1670 CLS                    :REM SCHERM WISSEN
1680 PRINT "*** HUISHOUDBOEKJE ***"
1690 PRINT "REKENINGNUMMER :";G;B$(G)
1700 PRINT
1710 PRINT " DATUM    OMSCHRIJVING               BEDRAG"
1720 IF P > -1 THEN 1740
1730 PRINT TAB(30)"----------"
1740 IF D = 0 THEN 1760
1750 PRINT D;                       :REM DATUM
1760 PRINT TAB(8) LEFT$(A$,20);          :REM OMSCHRIJVING
1770 REM ** POSITIONEREN BEDRAG **
1780 GOSUB 1240
1790 PRINT TAB(30+L)A$                  :REM BEDRAG
1800 P = P + 1
1810 IF P > 0 AND P < 8 THEN 1860
1820 PRINT TAB(30)"=========="
1830 REM ** WACHTEN **
1840 GOSUB 1890
1850 P = 0
1860 RETURN
1870 :
1880 REM *** WACHTEN ***
1890 PRINT "WILT U DOORGAAN <JA/NEE> ";
1900 INPUT A$
1910 A$ = LEFT$(A$,1)
1920 RETURN
1930 :
1940 REM **** STUURMODULE ****
1950 REM *** INITIALISEREN ***
1960 GOSUB 60
1970 REM *** VOORBEWERKING ***
1980 GOSUB 770
1990 REM *** INVOER ***
2000 GOSUB 180 190
2010 IF K = 0 THEN 2130
2020 IF K > 2 THEN 2050
2030 REM *** BEWERKING-BEREKENING ***
2040 GOSUB 1390
2050 IF K < 3 THEN 1980
2060 REM *** BEWERKING-TOTALEN ***
2070 GOSUB 900
2080 REM *** UITVOER TOTALEN ***
2090 GOSUB 980 990
2100 REM *** WACHTEN ***
2110 GOSUB 1890
2120 IF A$ = "J" THEN 1980
2130 END
2140 :
2150 :
2160 REM *** DATAREGELS ***
2170 REM *FORMAT REKENINGNR,DATUM,OMSCHRIJV.,BEDRAG *
2180 DATA 1,830103,"EDAH", 35.34
```

```
2190 DATA 9,830112,"CADEAU  ANN",12.75
2200 DATA 6,830131,HYPOTHEEK,1100
2210 DATA 7,830213,"JAS PETER",65
2220 DATA 7,830216,"BROEK MARTINE",60.05
2230 DATA 5,830215,GEB,325.75
2240 DATA 8,830228,"COST ACCOUNTING",75.05
2250 DATA 4,830217,BRANDVERZEKERING,120
2260 DATA 999
```

Voorbeeld

```
*** HUISHOUDBOEKJE ***

0 = STOPPEN
1 = REKENING OVER EEN BEPAALDE MAAND
2 = REKENING T/M GEBOEKTE DATUM
3 = TOTALEN OVER EEN BEPAALDE MAAND
4 = TOTALEN T/M GEBOEKTE DATUM

WAT IS UW KEUZE  4


*** HUISHOUDBOEKJE ***
T/M GEBOEKTE MAAND

 1 LEVENSMIDDELEN          35,34  2    %
 2 VAKANTIE / UITGAAN        -    0    %
 3 AUTO/MOTOR/FIETS          -    0    %
 4 VERZEKERINGEN          120,--  6.7  %
 5 ENERGIE EN WATER       325,75  18.2 %
 6 WOONLASTEN            1100,--  61.3 %
 7 KLEDING                125,05  7    %
 8 STUDIE EN OVERIGEN      75,05  4.2  %
 9 OVERIGEN                12,75  .7   %
                     -------------
   TOTAAL                1793,94  100%
WILT U DOORGAAN <JA/NEE> NEE
```

4.5 Hypotheekvergelijking

In een tijd waarin de hypotheekrentes onder druk staan, kan het interessant zijn van hypotheek te wisselen. Vaak is dit een oneindig rekenwerk dat zonder computer best te doen valt, maar dan wel veel tijd vergt.
Dit programma vraagt systematisch een aantal gegevens over de oude hypotheek en vraagt daarna een aantal gegevens over de nieuwe hypotheek. Er wordt bij dit programma van uitgegaan dat men een normale annuïteitenlening heeft.
Het resultaat van de berekening is de nieuwe annuïteit, de bruto maandelijkse last, het eventuele bruto voordeel en de eventuele terugverdientijd. Het resultaat is slechts een benadering, dat betekent dat de berekeningen niet exact zijn. De terugverdientijd is een indicatie of het zinvol is om een andere hypotheek te nemen. De terugverdientijd is de periode die nodig is om het totaal van kosten te dekken met het voordeel per maand. Mocht het voordeel negatief zijn, dan wordt de terugverdientijd niet berekend. Verder dient men er op bedacht te zijn dat er vrij weinig beveiligingen in het programma verwerkt zijn om foutieve invoer te ondervangen. Dit is gedaan omdat het anders veel te lang zou worden. Men kan dit er zelf alsnog in bouwen.
Verder dient men na te gaan of de tarieven in de interne data-file nog wel up to date zijn. Er is een aantal manieren om dit te controleren. Eén manier is om een makelaar te bellen. De andere manier is contact opnemen met de Vereniging Eigen Huis. Dit laatste geldt uitsluitend voor leden van de vereniging.
Aan de hand van een voorbeeld zal het programma worden uitgelegd. Albert en Marijke de Vries uit Voorschoten hebben twee jaar geleden een Premie B-woning gekocht met gemeentegarantie. De hypothecaire lening is 170 000 gulden groot, tegen een rentepercentage van 9,9% maandelijks achteraf. De looptijd is 30 jaar. Zij betalen aan maandelijkse annuïteit f 1478,–. Wanneer zij de lening in één keer aflossen moeten zij drie maanden boete betalen. In de algemene voorwaarden staat dat zij jaarlijks op de vervaldatum 15% boetevrij mogen aflossen.
De nieuwe lening, die zij kunnen krijgen is weer 170 000 gulden groot, maar tegen een rentepercentage van 8,2% tien jaar vast, te betalen maandelijks achteraf. Zij hebben geen taxatiekosten, omdat de gemeentegarantie naar de nieuwe hypotheekgever wordt overgeheveld. Albert en Marijke doen geen extra aflossingen, alleen moeten zij wel 1% afsluitprovisie betalen. Het geld om de boete en de kosten te betalen hebben zij.
Albert en Marijke zullen de nieuwe hypotheek accepteren, wanneer de terugverdientijd lager ligt dan drie jaar. De rentevastperiode is namelijk 10 jaar. De vraag is: nemen zij de nieuwe hypotheek ja of nee?
Het antwoord is JA. Bekijk de resultaten van de berekening maar!

Programma

```
10 REM *** HYPOTHEEK-VERGELIJKING ***
20 REM -- DATA BEDRAG/1000, NOTARISK., TAXATIEK. --
30 DATA 50,904,118, 60,957,123.90, 70,1010,144.55
40 DATA 80,1063,165.20, 90,1116,185.85
50 DATA 100,1269,206.50, 110,1334,227.15
60 DATA 120,1399,247.80, 130,1464,268.45
70 DATA 140,1528,289.10, 150,1593,309.75
80 DATA 160,1658,330.40, 170,1723,351.05
90 DATA 180,1787,371.70, 190,1852,392.35
100 DATA 200,1917,413, 210,1982,433.65
110 DATA 220,2046,454.30, 230,2111,474.95
120 DATA 240,2176,495.60, 0,0,0
130 DATA 38           :REM ROYEMENTSKOSTEN
140 DATA "TARIEVEN PER 15 APRIL 1982"
150 DIM A$(4)
160 GOTO 2220
170 :
180 REM *** INITIALISEREN ***
190 A$(1) = " HALFJAAR "
200 A$(2) = " MAAND "
210 A$(3) = " PROCENT "
220 A$(4) = " MAANDEN "
230 RETURN
240 :
250 REM *** FINANCIELE GEGEVENS I ***
260 REM ** GEGEVENS OUDE HYPOTHEEK **
270 CLS
280 PRINT "*** HYPOTHEEK-VERGELIJKING ***"
290 PRINT "-- GEGEVENS OVER HUIDIGE HYPOTHEEK --"
300 PRINT
310 PRINT "WAT IS HET RESTANT SCHULDBEDRAG    ";
320 INPUT H1
330 PRINT "TEGEN WELK PERCENTAGE              ";
340 INPUT R1
350 PRINT "HALFJAARLIJKS = 1 MAANDELIJKS = 2 ";
360 INPUT K1
370 IF K1 < 1 OR K1 > 2 THEN 350
380 M1 = 6
390 IF K1 = 1 THEN 410
400 M1 = 1
410 PRINT "BEDRAG AAN AFLOSSING + RENTE PER";A$(K1);" ";
420 INPUT B1
430 PRINT "BOETE BEREKEND O.B.V. PERC. = 1 MND = 2  ";
440 INPUT T1
450 IF T1 < 1 OR T1 > 2 THEN 430
460 PRINT "HOEVEEL"; A$(2+T1) ; "BOETE              ";
470 INPUT L1
480 PRINT "BOETE OVER HOEVEEL PROCENT V.D. SCHULD ";
490 INPUT Q1
500 RETURN
510 :
520 REM *** FINANCIELE GEGEVENS II ***
530 REM ** NIEUWE HYPOTHEEK **
```

```
540 CLS                      :REM SCHERM WISSEN
550 PRINT "*** HYPOTHEEK-VERGELIJKING ***"
560 PRINT "-- GEGEVENS OVER NIEUWE HYPOTHEEK --"
570 PRINT
580 PRINT "HOEVEEL WILT U EXTRA AFLOSSEN ";
590 INPUT H3
600 PRINT "WAT WORDT DE LOOPTIJD ";
610 INPUT N2
620 PRINT "WAT IS HET NIEUWE RENTE-PERC. ";
630 INPUT R2
640 PRINT "BETALING PER HALFJ = 1 PER MND = 2 ";
650 INPUT K2
660 IF K2 < 1 OR K2 > 2 THEN 640
670 M2 = 6
680 IF K2 = 1 THEN 700
690 M2 = 1
700 PRINT "HOEVEEL PROC. AFSLUITPROVISIE ";
710 INPUT A2
720 PRINT "TAXATIE-KOSTEN  JA = 1 NEE = 2 ";
730 INPUT D2
740 IF D2 < 1 OR D2 > 2 THEN 720
750 RETURN
760 :
770 REM *** BOETEBERKENING O.B.V. MND ***
780 X = H1 / 100 * Q1 / 100
790 B = L1 / 12 * X  * R1
800 B = INT(B * 100 + 0.5)/100
810 RETURN
820 :
830 REM *** BOETEBEREKENING O.B.V. PERC ***
840 X = H1 / 100 * Q1 / 100
850 B = L1 * X
860 B = INT(B * 100 + 0.5)/100
870 RETURN
880 :
890 REM *** PROVISIEKOSTEN ***
900 V = A2/100 * (H1 - H3)
910 RETURN
920 :
930 REM *** NOTARIS- EN TAXATIEKOSTEN ***
940 RESTORE
950 X = -1
960 IF X = -1 THEN 1000
970 X1 = X
980 Y1 = Y
990 Z1 = Z
1000 READ X,Y,Z
1010 X = X * 1000
1020 IF X = 0 AND Z = 0 THEN 1160
1030 IF H1 - H3  > X THEN 960
1040 J = Y
1050 P = Z
1060 IF H1 = X THEN 1110
1070 J = (Y - Y1)/10000
1080 P = (Z - Z1)/10000
```

```
1090 J = Y1 + INT(J * ((H1-H3) - X1) + 0.5) :REM NOTARIS
1100 P = Z1 + INT(P * ((H1-H3) - X1) + 0.5) :REM TAXATIE
1110 IF D2 = 1 THEN 1140
1120 P = 0
1130 REM ** DUMMY LEZEN **
1140 READ X,Y,Z
1150 IF X <> 0 AND Z <> 0 THEN 1140
1160 READ I,D$             :REM ROYEMENTSK. + TEKST
1170 RETURN
1180 :
1190 REM *** NIEUWE ANNUITEIT ***
1200 M = 2
1210 IF K = 1 THEN 1230
1220 M = 12
1230 A = (R/M)/100
1240 X = (1+A)↑(N2 * M)
1250 S = 1 - 1/X
1260 S = (H * A)/S
1270 RETURN
1280 :
1290 REM *** KOSTENBEREKENING ***
1300 REM *** BOETE ***
1310 ON T1 GOSUB 840  ,780
1320 REM *** PROVISIE ***
1330 GOSUB 900
1340 REM *** NOTARIS + TAXATIEKOSTEN ***
1350 GOSUB 940
1360 F = B + V + J + P + I        :REM TOT.KOSTEN
1370 RETURN
1380 :
1390 REM *** VOORDEEL OMSLUITEN ***
1400 H = H1 - H3
1410 R = R2
1420 K = K2
1430 GOSUB 1200
1440 B2 = S
1450 G = B1/M1 - B2/M2 :REM VOORDEEL PER MND
1460 RETURN
1470 :
1480 REM *** MAANDENTERUGVERDIEN TIJD ***
1490 IF G < 0 THEN 1550
1500 E = F/G
1510 IF INT(F/G) - E = 0 THEN 1550
1520 E = INT(E + 1)
1530 E1 = INT(E/12)
1540 E2 = E - E1 * 12
1550 RETURN
1560 :
1570 :
1580 REM *** POSITIONEREN BEDRAG ***
1590 B$ = "--"
1600 W = Z - INT(Z)
1610 A$ = STR$(INT(Z))
1620 IF W = 0 THEN 1670
1630 W = INT(W * 100 + 0.5)
```

```
1640 B$ = STR$(W)
1650 B$ = "00" + RIGHT$(B$,LEN(B$)-1)
1660 B$ = RIGHT$(B$,2)
1670 A$ = A$ + "," + B$
1680 IF Z <> 0 THEN 1700
1690 A$ = "-   "
1700 Z = 10 - LEN(A$)
1710 RETURN
1720 :
1730 REM *** WACHTEN ***
1740 PRINT "WILT U DOORGAAN <JA/NEE> ";
1750 INPUT D$
1760 D$ = LEFT$(D$,1)
1770 RETURN
1780 :
1790 REM *** UITVOER ***
1800 CLS
1810 PRINT "*** HYPOTHEEK-VERGELIJKING ***"
1820 PRINT
1830 PRINT "- KOSTEN : ";D$
1840 PRINT
1850 Z = B : GOSUB 1590
1860 PRINT "1. BOETE...........:";TAB(28+Z)A$
1870 Z = J : GOSUB 1590
1880 PRINT "2. NOTARISKOSTEN...:";TAB(28+Z)A$
1890 Z = P : GOSUB 1590
1900 PRINT "3. TAXATIEKOSTEN...:";TAB(28+Z)A$
1910 Z = I : GOSUB 1590
1920 PRINT "4. ROYEMENTSKOSTEN.:";TAB(28+Z)A$
1930 Z = V : GOSUB 1590
1940 PRINT "5. PROVISIEKOSTEN..:";TAB(28+Z)A$
1950 Z = F : GOSUB 1590
1960 PRINT TAB(26) "------------"
1970 PRINT "TOTALE KOSTEN......:";TAB(28+Z)A$
1980 PRINT TAB(26) "============"
1990 GOSUB 1740
2000 IF D$ = "N" THEN 2180
2010 CLS                    :REM SCHERM WISSEN
2020 PRINT "*** HYPOTHEEK VERGELIJKING ***"
2030 PRINT
2040 PRINT "==> OPNIEUW AFSLUITEN VAN DE HYPOTHEEK <=="
2050 PRINT
2060 Z = H1 - H3 : GOSUB 1590
2070 PRINT "NIEUW HYPOTHEEKBEDRAG :";TAB(29+Z)A$
2080 PRINT
2090 Z = B2 : GOSUB 1590
2100 PRINT "NIEUWE ANNUITEIT PER"+A$(K2);TAB(29+Z)A$
2110 Z = B2/M2 : GOSUB 1590
2120 PRINT "BRUTO-LAST PER MAAND :";TAB(29+Z)A$
2130 Z = G : GOSUB 1590
2140 PRINT "VOORDEEL PER MAAND:";TAB(29+Z)A$
2150 IF G < 0 THEN 2170
2160 PRINT "TERUGVERDIENTIJD ";E1;"JAAR EN ";E2;"MAANDEN"
2170 PRINT
2180 RETURN
```

```
2190 :
2200 REM **** STUURMODULE ****
2210 REM *** INITIALISEREN ***
2220 GOSUB 190
2230 REM *** FINANCIELE GEGEVENS I ***
2240 GOSUB 270
2250 REM *** FINANCIELE GEGEVENS II ***
2260 GOSUB 540
2270 REM *** KOSTENBEREKENING ***
2280 GOSUB 1310
2290 REM *** VOORDEEL/ANNUITEIT ***
2300 GOSUB 1400
2310 REM *** TERUGVERDIENTIJD ***
2320 GOSUB 1490
2330 REM *** UITVOER ***
2340 GOSUB 1800
2350 END
```

Voorbeeld

```
*** HYPOTHEEK-VERGELIJKING ***
-- GEGEVENS OVER HUIDIGE HYPOTHEEK --

WAT IS HET RESTANT SCHULDBEDRAG     170000
TEGEN WELK PERCENTAGE               9.9
HALFJAARLIJKS = 1 MAANDELIJKS = 2  2
BEDRAG AAN AFLOSSING + RENTE PER MAAND   1478
BOETE BEREKEND O.B.V. PERC. = 1 MND = 2   2
HOEVEEL MAANDEN BOETE                3
BOETE OVER HOEVEEL PROCENT V.D. SCHULD  85


*** HYPOTHEEK-VERGELIJKING ***
-- GEGEVENS OVER NIEUWE HYPOTHEEK --

HOEVEEL WILT U EXTRA AFLOSSEN  0
WAT WORDT DE LOOPTIJD  30
WAT IS HET NIEUWE RENTE-PERC.  8.2
BETALING PER HALFJ = 1 PER MND = 2  2
HOEVEEL PROC. AFSLUITPROVISIE  1
TAXATIE-KOSTEN  JA = 1 NEE = 2  2


*** HYPOTHEEK-VERGELIJKING ***

- KOSTEN : TARIEVEN PER 15 APRIL 1982

1. BOETE...........:               3576,37
2. NOTARISKOSTEN...:               1723,--
3. TAXATIEKOSTEN...:                  -
4. ROYEMENTSKOSTEN.:                 38,--
5. PROVISIEKOSTEN..:               1700,--
                               ------------
TOTALE KOSTEN......:               7037,37
                               ============
WILT U DOORGAAN <JA/NEE> JA

*** HYPOTHEEK VERGELIJKING ***

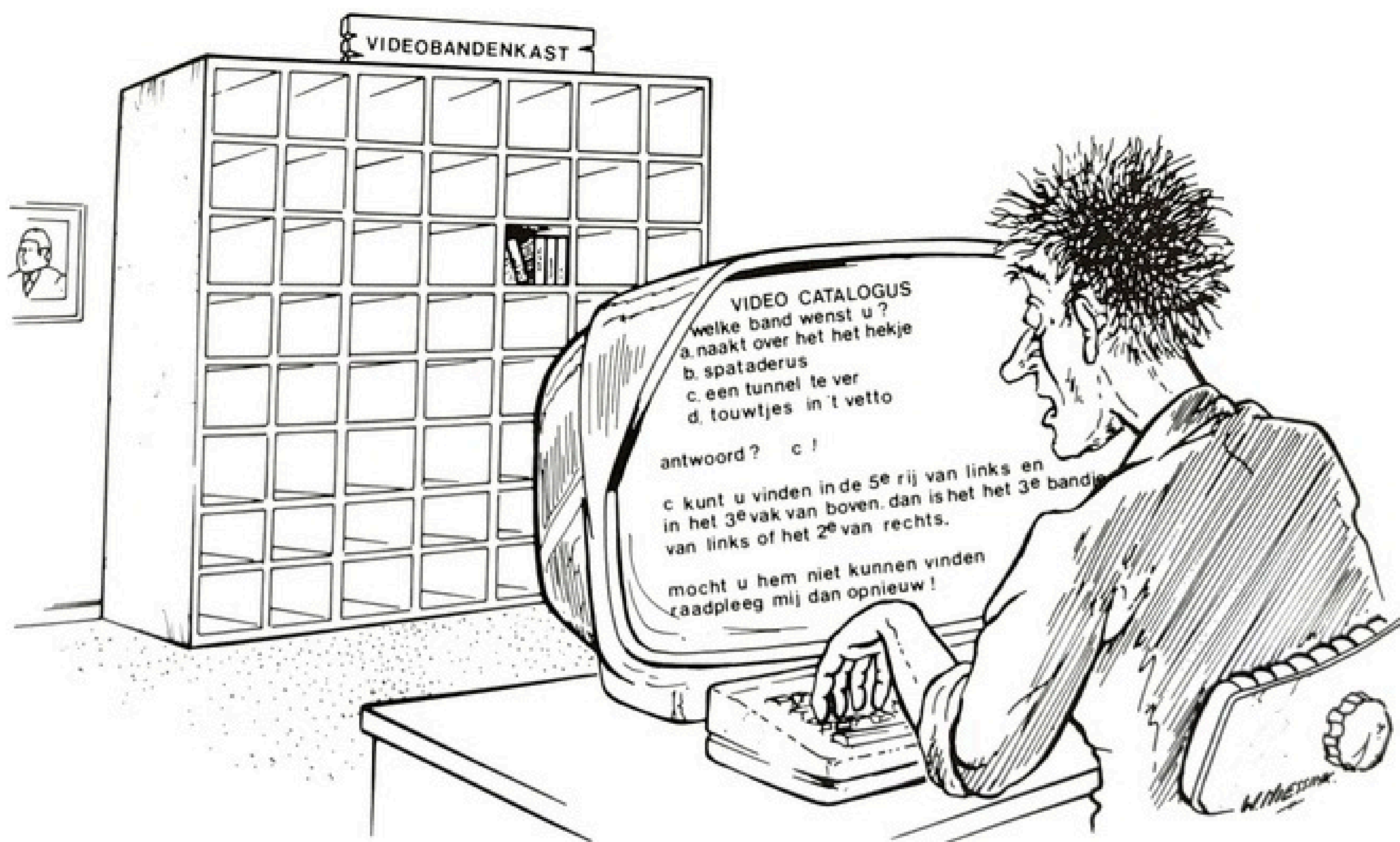
==> OPNIEUW AFSLUITEN VAN DE HYPOTHEEK <==

NIEUW HYPOTHEEKBEDRAG :         170000,--

NIEUWE ANNUITEIT PER MAAND        1271,18
BRUTO-LAST PER MAAND :            1271,18
VOORDEEL PER MAAND:                206,82
TERUGVERDIENTIJD  2 JAAR EN  11 MAANDEN
```

4.6 Catalogus Video

Dit catalogusprogramma maakt het mogelijk te zoeken in grote hoeveelheden gegevens. Men moet zich eens indenken dat men een bepaalde film wil zien, maar men niet meer weet op welke band de film staat. Dit programma biedt de gelegenheid op dit soort vragen razendsnel een antwoord te geven, zonder dat men zelf daadwerkelijk hoeft te zoeken.
Aan de hand van een voorbeeld zal worden uitgelegd hoe het programma werkt.
Stel men wil weten of er op band 1 films staan met Roger Moore als acteur. Wanneer het programma gestart is, zal het een zestal rubrieken laten zien waaruit men kan kiezen: titel, acteur/actrice, band, start teller en einde teller. Bij de vraag 'Geef rubriek of nummer' geeft men het nummer van de corresponderende rubriek welke men wil kiezen of men toetst gewoon de rubriekbenaming in. In dit geval wordt BAND ingetoetst. Het programma komt na acceptatie direct terug met de volgende vraag 'Waaraan dient BAND te voldoen?' In het voorbeeld wil men weten of op band 1 films voorkomen met Roger Moore. Men toetst dus een 1 in. Het programma zal de ingevoerde voorwaarde op het beeldscherm zetten en zal tegelijkertijd om de volgende voorwaarden vragen. Om de verschillende voorwaarden aan elkaar te koppelen waaraan de selecties moeten voldoen wordt gebruik gemaakt van EN of OF. Heeft men geen voorwaarden meer dan geeft men 0, ten teken dat het programma met het selectieproces kan beginnen. In het gegeven voorbeeld dient aan beide voorwaarden voldaan te worden. De band moet 1 zijn EN de acteur Roger Moore. Er wordt dus een 2 ingevoerd, ten teken dat het een EN situatie is. Nu herhaalt zich weer dezelfde cyclus als hiervoor beschreven. Men geeft bij rubriek op dat men een selectie wil maken op de rubriek acteur/actrice. Deze rubriek dient te voldoen aan Roger Moore. Het is voldoende om alleen Roger in te voeren. Het programma kijkt uit hoeveel karakters de voorwaarde bestaat en neemt uit het bestand de gevraagde rubriek en neemt precies evenveel karakters op om te vergelijken. Ten slotte wordt opgegeven dat men stoppen wil door bij de vraag 'OF = 1 EN = 2 STOP = 0' een nul te geven.



Zodra de nul is ingedrukt, start het zoekproces. Voldoet een record dan wordt het op het beeldscherm gezet. Het kan zijn dat er meer records zijn. Het is ook mogelijk het zoekproces te onderbreken en een nieuwe selectie in te voeren.
Wanneer men de interne data-file vult met gegevens in hoofdletters dan moeten de condities ook in hoofdletters worden ingevoerd. Gebeurt dit niet dan zal het systeem geen enkele overeenkomst vinden. Dit komt doordat de computer hoofdletters en kleine letters als twee totaal verschillende karakters beschouwt. Het zou zeer handig zijn als het systeem er zelf voor zou zorgen dat het geen verschil maakt tussen hoofdletters en kleine letters. Dit probleem is oplosbaar. Het enige dat men moet doen, is een subroutine toevoegen. Het gaat om de subroutine '*** ALFANUMERIEK ***'. Er wordt in deze subroutine van uitgegaan dat de ASCII-waarde van A gelijk is aan 65 en de ASCII-waarde van a gelijk is aan 97. De subroutine moet er dan als volgt uitzien.

```
1660 Z = LEN(B$(I)) : E$ = "" : F$ = ""
1661 FOR Y = 1 TO Z
1662 A$ = LEFT$(B$(I),Y)
1663 F = ASC(RIGHT$(A$,1))
1664 IF F > 122 OR F < 97 THEN 1666
1665 F = F - 32
1666 E$ = E$ + CHR$(F)
1667 A$ = LEFT$(C$(K,2),Y)
1668 F = ASC(RIGHT$(A$,1))
1669 IF F > 122 OR F < 97 THEN 1771
1670 F = F - 32
1671 F$ = F$ + CHR$(F)
1672 NEXT Y
1673 IF E$ <> F$ THEN 1690
1680 V = 1
1690 RETURN
```

Deze catalogus is ingericht voor videocassettes, maar het is vrij eenvoudig deze catalogus ook geschikt te maken voor muziekcassettes.
De interne data-file spreekt voor zich. Een DATA-regel is opgebouwd uit: titel, acteur/actrice, genre, bandnummer, startteller en eindteller. De laatste drie rubrieken zijn numerieke gegevens. De eerste drie zijn alfanumeriek, die moeten tussen aanhalingstekens. De laatste DATA-regel moet '999' bevatten. Deze regel geeft het einde van de file aan (zie regel 2160).

Programma

```
10 REM *** VIDEO-CATALOGUS ***
20 DIM A$(6), B(10,3), B$(10), C(3,2), C$(3,2)
30 GOTO 1910
40 :
50 REM *** INITIALISEREN ***
60 A$(1) = "TITEL"
70 A$(2) = "ACTEUR/ACTRICE"
80 A$(3) = "GENRE"
90 A$(4) = "BAND"
100 A$(5) = "START-TELLER"
110 A$(6) = "EIND-TELLER"
```

```
120 RETURN
130 :
140 REM *** INVOER ***
150 REM ** KOP AFDRUKKEN **
160 GOSUB 900
170 FOR I = 1 TO 3
180 PRINT I;TAB(3)"= "; A$(I);
190 PRINT TAB(21)I+3;TAB(24)"= ";A$(I+3)
200 NEXT I
210 REM ** INGEGEVEN VOORWAARDEN AFDRUKKEN **
220 GOSUB 960
230 REM ** VOORWAARDEN **
240 GOSUB 290
250 IF H = 1 THEN 160
260 RETURN
270 :
280 REM *** VOORWAARDEN ***
290 H = 0                          :REM STUURVARIABELE
300 X = T + 1
310 IF X > 10 THEN 570
320 B(X,1) = 1                     :REM OF=1 EN=2 !
330 IF X = 1 THEN 380
340 PRINT "OF = 1 EN = 2 STOP = 0 ";
350 INPUT B(X,1)
360 IF B(X,1) < 0 OR B(X,1) > 2 THEN 340
370 IF B(X,1) = 0 THEN 570
380 PRINT "GEEF RUBRIEK-NAAM OF NUMMER ";
390 INPUT A$
400 A$ = LEFT$(A$,1)
410 B(X,2) = VAL(A$)
420 IF B(X,2) > 0 AND B(X,2) < 7 THEN 510
430 REM ** ZOEKEN NAAM RUBRIEK **
440 FOR I = 1 TO 6
450 IF LEFT$(A$(I),1) <> A$ THEN 480
460 B(X,2) = I
470 I = 6               :REM MAXIMUM LUS
480 NEXT I
490 IF B(X,2) = 0 THEN 380
500 REM ** CONDITIE **
510 PRINT "WAARAAN DIENT ";A$(B(X,2));" TE VOLDOEN ";
520 INPUT B$(X)
530 IF B(X,2) < 4 THEN 550
540 B(X,3) = VAL(B$(X))
550 T = X
560 H = 1
570 RETURN
580 :
590 REM *** VOORBEWERKING ***
600 T = 0
610 Y = 1
620 RESTORE             :REM DATAPOINTER VOORAAN
630 FOR I = 1 TO 10
640 FOR J = 1 TO 3
650 B(I,J) = 0
660 NEXT J
```

```
670 B$(I) = ""
680 NEXT I
690 P = 0                            :REM AANTAL RECORDS
700 REM ** PRINT MAT SCHOONVEGEN **
710 FOR I = 1 TO 3
720 FOR J = 1 TO 2
730 C(I,J) = 0
740 C$(I,J) = ""
750 NEXT J
760 NEXT I
770 RETURN
780 :
790 REM *** RECORD LEZEN ***
800 REM ** UIT INTERNE DATA-FILE **
810 H = 0                           :REM EOF ?
820 READ C$
830 IF C$ = "999" THEN 870
840 C$(1,2) = C$
850 H = 1                           :REM NOT(EOF)
860 READ C$(2,2),C$(3,2),C(1,2),C(2,2),C(3,2)
870 RETURN
880 :
890 REM *** KOP AFDRUKKEN ***
900 CLS                     :REM SCHERM WISSN
910 PRINT "*** CATALOGUS - VIDEO ***
920 PRINT
930 RETURN
940 :
950 REM *** VOORWAARDEN AFDRUKKEN ***
960 IF T = 0 THEN 1070
970 PRINT "REEDS OPGENOMEN VOORWAARDEN !"
980 FOR I = 1 TO T
990 A$ = " "
1000 IF I = 1 THEN 1040
1010 A$ = "EN"
1020 IF B(I,1) = 2 THEN 1040
1030 A$ = "OF"
1040 PRINTI; TAB(4)A$;TAB(7)A$(B(I,2));
1050 PRINT TAB(24)" = ";B$(I)
1060 NEXT I
1070 RETURN
1080 :
1090 REM *** BEWERKING ***
1100 REM ** RECORD LEZEN **
1110 GOSUB 810
1120 IF H = 0 THEN 1320
1130 R = 0                  :REM STUURVARIABELE
1140 REM ** VOORWAARDEN TOETSEN **
1150 FOR I = 1 TO T
1160 K = B(I,2)             :REM RUBRIEKNUMMER
1170 V = 0                  :REM ** STUURVARIABELE
1180 L = 1                  :REM ALFANUMERIEK
1190 IF K < 4 THEN 1210
1200 L = 2
1210 ON L GOSUB 1660 ,1720
```

```
1220 S = R + V
1230 R = 1                  :REM VOLDOET
1240 IF S >= B(I,1) THEN 1260
1250 R = 0                  :REM VOLDOET NIET
1260 NEXT I
1270 IF R = 0 THEN 1320
1280 REM ** REGEL AFDRUKKEN **
1290 GOSUB 1360
1300 REM ** SCHUIVEN **
1310 GOSUB 1570
1320 RETURN
1330 :
1340 REM *** UITVOER ***
1350 REM ** KOP AFDRUKKEN **
1360 GOSUB 900
1370 P = P + 1
1380 PRINT "AANTAL GEACCEPTEERDE RECORDS ";P
1390 PRINT
1400 REM ** AFDRUK **
1410 FOR J = 1 TO 2
1420 IF C(1,J) = 0 AND C(3,J) = 0 THEN 1490
1430 FOR I = 1 TO 3
1440 PRINT A$(I);TAB(19)": ";C$(I,J)
1450 NEXT I
1460 FOR I = 1 TO 3
1470 PRINT A$(I+3);TAB(19)":";C(I,J)
1480 NEXT I
1490 NEXT J
1500 PRINT
1510 PRINT "ER KUNNEN MEER RECORDS VOLDOEN !"
1520 Y = 1
1530 GOSUB 1770
1540 RETURN
1550 :
1560 REM *** OPSCHUIVEN ***
1570 FOR I = 1 TO 3
1580 C$(I,1) = C$(I,2)
1590 C(I,1) = C(I,2)
1600 C$(I,2) = ""
1610 C(I,2) = 0
1620 NEXT I
1630 RETURN
1640 :
1650 REM *** ALFANUMERIEK ? ***
1660 Z = LEN(B$(I))
1670 IF B$(I) <> LEFT$(C$(K,2),Z) THEN 1690
1680 V = 1                  :REM VOLDOET
1690 RETURN
1700 :
1710 REM *** NUMERIEK ? ***
1720 IF B(I,3) <> C(K-3,2) THEN 1740
1730 V = 1                  :REM VOLDOET
1740 RETURN
1750 :
1760 REM *** WACHTEN ***
```

```
1770 A$ = "WILT U DOORGAAN"
1780 IF Y = 1 THEN 1800
1790 A$ = "WILT U NOG EEN SELECTIE"
1800 PRINT A$;"  <JA/NEE> ";
1810 INPUT A$
1820 A$ = LEFT$(A$,1)
1830 Y = 1                    :REM DOORGAAN
1840 IF A$ = "J" THEN 1870
1850 Y =  0
1860 IF A$ <> "N" THEN 1770
1870 RETURN
1880 :
1890 REM **** STUURMODULE ****
1900 REM *** INITIALISEREN ***
1910 GOSUB 60
1920 REM *** VOORBEWERKING ***
1930 GOSUB 600
1940 REM *** INVOER ***
1950 GOSUB 160
1960 REM *** BEWERKING ***
1970 GOSUB 1110
1980 IF H <> 0 AND Y <> 0 THEN 1970
1990 REM *** NIEUWE SELECTIE ***
2000 Y = 2
2010 GOSUB 1770
2020 IF A$ = "J" THEN 1930
2030 END
2040 :
2050 REM ----------------------------------------------------
2060 :
2070 REM ***** INTERNE DATA-FILE *****
2080 DATA "SCHONE SLAAPSTER","GRETA GARBO",MUSICAL
2090 DATA 1,0,1200
2100 DATA "KING KONG","WILLIAM HOLDEN",ACTIE
2110 DATA 1,1201,1700
2120 DATA "FINAL COUNT DOWN","KIRK DOUGLAS",OORLOG
2130 DATA 3,0,1200
2140 DATA "YOU ONLY LIVE TWICE","ROGER MOORE",ACTIE
2150 DATA 1,1701,3500
2160 DATA "999"
```

Voorbeeld

```
*** CATALOGUS - VIDEO ***

 1 = TITEL                  4 = BAND
 2 = ACTEUR/ACTRICE         5 = START-TELLER
 3 = GENRE                  6 = EIND-TELLER
GEEF RUBRIEK-NAAM OF NUMMER BAND
WAARAAN DIENT BAND TE VOLDOEN 1


*** CATALOGUS - VIDEO ***

 1 = TITEL                  4 = BAND
 2 = ACTEUR/ACTRICE         5 = START-TELLER
 3 = GENRE                  6 = EIND-TELLER
REEDS OPGENOMEN VOORWAARDEN !
 1      BAND                   = 1
OF = 1 EN = 2 STOP = 0  2
GEEF RUBRIEK-NAAM OF NUMMER 2
WAARAAN DIENT ACTEUR/ACTRICE TE VOLDOEN ROGER MOORE


*** CATALOGUS - VIDEO ***

 1 = TITEL                  4 = BAND
 2 = ACTEUR/ACTRICE         5 = START-TELLER
 3 = GENRE                  6 = EIND-TELLER
REEDS OPGENOMEN VOORWAARDEN !
 1      BAND                   = 1
 2   EN ACTEUR/ACTRICE         = ROGER MOORE
OF = 1 EN = 2 STOP = 0  0


*** CATALOGUS - VIDEO ***

AANTAL GEACCEPTEERDE RECORDS  1

TITEL                  : YOU ONLY LIVE TWICE
ACTEUR/ACTRICE         : ROGER MOORE
GENRE                  : ACTIE
BAND                   : 1
START-TELLER           : 1701
EIND-TELLER            : 3500

ER KUNNEN MEER RECORDS VOLDOEN !
WILT U DOORGAAN  <JA/NEE> JA
WILT U NOG EEN SELECTIE  <JA/NEE> NEE
```

Appendix

In deze appendix zijn de volgende conversietabellen opgenomen:
Tabel 1: String-functies.
Tabel 2: Rekenkundige en logische functies.
Tabel 3: Statements.
De tabellen kunnen worden gebruikt bij het aanpassen van programma's die op een ander type computer zijn geschreven dan de in uw bezit zijnde computer.
Het gebruik van de tabellen gaat als volgt:
In het aan te passen programma ziet u een statement waarvan u denkt dat die niet op uw computer voorkomt. U zoekt de statement op in de tabellen, controleert op welke computers die statement voorkomt, en indien hij inderdaad op uw computer niet voorkomt dan kunt u in de kolom 'overeenkomstige statements' vinden of er vervangende statements zijn. Zo ja, dan kunt u de voor uw computer geldende vervangende statement in het aan te passen programma gebruiken.
Statements en functies die alleen op één computer voorkomen, en waarvoor geen vervangende statement of functie bestaat, zijn in deze tabellen meestal niet opgenomen.
De auteur van dit boek heeft getracht de programma's zoveel mogelijk in converteerbare statements te schrijven.

Tabel 1. String-functies.

string-functies	BBC-computer	Commodore 64	APPLE II	DAI	EXIDY SORC.	PET/CBM	P2000	TRS80	ZX81	VIC20	ACORN ATOM	overeenkomstige string-functies
$ *n*											X	CHR$(*n*)
ASC(*str*)	X	X	X	X	X	X	X	X		X		CH
CH											X	ASC(*str*)
CHR$(*n*)	X	X	X	X	X	X	X	X	X	X		$ (*n*)
CODE									X			
FRE('')		X					X					FRE(*n*$)
FRE(*n*$)					X			X				FRE('')
GET	X	X	X	X		X				X		INKEY$
INKEY$	X							X	X			GET
LEFT$(*str,n*)	X	X	X	X	X	X	X	X		X	X	
LEN(*str*)	X	X	X	X	X	X	X	X	X	X	X	
MID$(*str,m,n*)	X	X	X	X	X	X	X	X		X	X	
RIGHT$(*str,n*)	X	X	X	X	X	X	X	X		X	X	
slicing									X			geeft LEFT$, MID$ en RIGHT$ functies
STR$(*n*)	X	X	X	X	X	X	X	X	X	X	X	
STRING$(*n,kar*)	X						X	X				
VAL(*str*)	X	X	X	X	X	X	X	X	X	X	X	
VARPTR *n*				X			X	X				

Tabel 2. Rekenkundige en logische functies.

functies	BBC-computer	Commodore 64	APPLE II	DAI	EXIDY SORC.	PET/CBM	P2000	TRS80	ZX81	VIC20	ACORN ATOM	overeenkomstige functies
& *n*	X										X	HEX$(*n*)
%											X	CINT, FIX
?FRE(Ø)			X									FRE
ABS	X	X	X	X	X	X	X	X	X	X	X	
ACS	X								X			
AND	X		X	X	X	X	X	X	X		X	
ASN	X								X			
ATN	X	X	X	X	X	X	X	X	X	X	X	
CALLM. . .,(*x*)				X								LINK *n*, USR(*x*)
CDBL							X	X				
CINT							X	X				%
COS *n*	X	X	X	X	X	X	X	X	X	X	X	
CSGN							X	X				
CUR				X								POS
DEG	X										X	
ERL	X						X	X				
ERR	X						X	X				
EXP *n*	X	X	X	X	X	X	X	X	X	X	X	
FIX							X	X				%
FLT											X	
FRAC				X								
FRE		X		X	X	X	X	X		X		?FRE(Ø)
HEX$(*n*)				X			X					& *n*
HTN											X	
INT	X	X	X	X	X	X	X	X	X	X		
LINK *n*											X	CALLM. . .,(*x*), USR(*x*)
LN	X								X			LOG
LOG	X	X	X	X	X	X	X	X		X	X	LN
LOGT				X								
MOD	X		X	X			X					*x*%*y*
NOT			X	X	X	X	X	X	X			
OCT$(*n*)							X					
OR	X		X	X	X	X	X	X	X		X	
POS	X	X	X		X	X	X	X				CUR
RAD	X										X	
RND	X			X							X	RND(Ø), RND(*neg*)
RND%*n*											X	
RND(Ø)					X	X	X	X	X	X		RND(*neg*), RND
RND (*n*)		X					X	X				RND(*pos*)
RND(*neg*)			X									RND(Ø), RND
RND(*pos*)			X									RND(*n*)
SCRN(*x*,*y*)			X	X								
SGN	X	X	X	X	X	X	X	X	X	X	X	
SIN *n*	X	X	X	X	X	X	X	X	X	X	X	
SPC *m*	X	X	X	X	X		X					
SPC *n*						X				X		
SQR(*n*)	X	X	X	X	X	X	X	X	X	X	X	
SWAP							X					
TAN(*n*)	X	X	X	X	X	X	X	X	X	X	X	
USR(*x*)	X	X	X		X	X	X	X	X	X		CALLM. . .,(*x*), LINK *n*
x%*y*											X	MOD

Tabel 3. Statements.

statements	BBC-computer	Commodore 64	APPLE II	DAI	EXIDY SORC.	PET/CBM	P2000	TRS80	ZX81	VIC20	ACORN ATOM	overeenkomstige statements
$											X	DEFSTR, IMPSTR
%											X	DEFDBL
?*m*=*n*	X										X	POKE *m*,*n*
?*n*	X										X	IN(*poort*), IN#*n*, INP, INP(*n*)
AT									X			CURSOR *m*,*n*
BEEP *x*,*y*											X	SOUND
CALL	X		X									CALLM, SYS, LINK
CALLM				X								CALL, SYS, LINK
CLEAR *n*	X										X	MODE *n*
CLEAR Ø	X										X	MODE Ø, TEXT, GR
CLOAD*							X					INPUT#*n*,*m*, RECALL, LOADA, LOAD, GET A
CLOSE	X	X				X				X		SHUT
CLS	X							X	X			HOME, PRINT $12
COLOR *n*											X	HCOLOR=*n*
COLOR=*n*			X									COLOUR *n*
COLOUR *n*	X										X	COLOR=*n*
COUNT	X										X	
COUNT *n*	X										X	TAB
CSAVE*							X					SAVE A, STORE
CURSOR *m*,*n*				X						X		AT
DATA	X	X	X	X	X	X	X	X		X	X	
DEF FN (*naam*)	X	X	X			X	X			X		
DEFDBL							X	X				%
DEFINT							X	X				IMPINT
DEFSNG							X	X				IMPFPT
DEFSTR								X				IMPSTR, $
DIM *var*(*n*)	X	X	X	X	X	X	X	X	X	X	X	
DO..UNTIL											X	
DOT				X								SET (*x*,*y*), PLOT, HPLOT, PLOT *k*,*x*,*y*
DRAW	X			X								
DSP *n*			X									
END	X	X	X	X	X	X	X	X		X	X	
ERROR	X						X	X				
EXT	X										X	
FILL				X								
FIN											X	OPEN
FOR..TO..STEP	X	X	X	X	X	X	X	X	X	X	X	
GET	X										X	
GET A	X	X									X	INPUT#*n*,*m*, RECALL, LOADA, CLOAD*, LOAD
GOSUB *n*	X	X	X	X	X	X	X	X	X	X	X	
GOTO *n*	X	X	X	X	X	X	X	X	X	X	X	
GR			X									CLEAR Ø
HCOLOR=*n*			X									COLOR *n*
HOME			X									CLS, PRINT $12
HPLOT			X									DOT, PLOT, PLOT *k*,*x*,*y*
HTAB (*n*)			X									
IF..GOSUB *n*	X		X	X	X	X	X	X			X	
IF..GOTO *n*	X		X	X	X	X	X	X		X	X	
IF..THEN..	X	X	X	X	X	X	X	X	X	X	X	
IF..THEN..ELSE	X						X	X			X	
IMPFPT				X								DEFSNG
IMPINT				X								DEFINT
IMPSTR				X								DEFSTR, $
IN (*poort*)					X		X					IN#*n*, INP, INP(*n*), ?*n*
IN#*n*			X									IN (*poort*), INP, INP(*n*), ?*n*
INP				X								IN (*poort*), IN#*n*, INP (*n*), ?*n*
INP (*n*)								X				IN (*poort*), IN#*n*, INP, ?*n*
INPUT *n*	X	X							X			INPUT "..",*n*
INPUT"..",*n*	X		X	X	X	X	X	X		X	X	INPUT *n*
INPUT#*n*,*m*		X			X	X		X		X		RECALL, LOADA, CLOAD*,LOAD, GET A
LET	X	X	X	X	X	X	X	X	X	X	X	

Tabel 3. Statements (vervolg).

statements	BBC-computer	Commodore 64	APPLE II	DAI	EXIDY SORC.	PET/CBM	P2000	TRS80	ZX81	VIC20	ACORN ATOM	overeenkomstige statements
LINK											X	CALL, CALLM, SYS
LOAD	X								X			INPUT#*n*,*m*, RECALL, LOADA, CLOAD*, GET A
LOADA				X								INPUT#*n*,*m*, RECALL, CLOAD*, LOAD, GET A
LPRINT							X	X	X			PRINT $2, PRINT=
MODE *n*	X			X								CLEAR *n*
MODE Ø				X								TEXT, CLEAR Ø
MOVE	X										X	
NEXT	X	X	X	X	X	X	X	X	X	X	X	
ON..ERROR GOTO *n*	X						X	X			X	ONERR
ON..GOSUB *n*	X	X	X	X	X	X	X	X		X	X	
ON..GOTO *n*	X	X	X	X	X	X	X	X		X	X	
ONERR			X									ON..ERROR GOTO *n*
OPEN		X				X				X		FIN
OUT (*poort, n*)				X			X	X				PR#*n*, PRINT#*n*
PAUSE									X			WAIT
PDL			X	X								
PEEK (*n*)		X	X	X	X	X	X	X	X	X		PRINT ?*n*
PLOT			X						X			SET (*x,y*), HPLOT, DOT, PLOT *k,x,y*
PLOT *k,x,y*	X										X	RESET (*x,y*), UNPLOT, SET (*x,y*), HPLOT, DOT
POINT	X							X				
POKE *m,n*		X	X	X	X	X	X	X	X	X		?*m,n*
PR#*n*			X				X	X				OUT (*poort, n*), PRINT#*n*
PRINT	X	X	X	X	X	X	X	X	X	X	X	
PRINT $12											X	CLS, HOME
PRINT $2											X	PRINT=, LPRINT
PRINT=						X				X		SAVEA, STORE, LPRINT
PRINT ?*n*	X										X	PEEK (*n*)
PRINT @*n*								X				
PRINT USING							X	X				
PRINT#n	X	X				X				X		OUT (*poort, n*), PR#*n*
PTR (*n*)	X										X	
PUT											X	
READ	X	X	X	X	X	X	X	X		X	X	
RECALL			X									INPUT#*n*,*m*, LOADA, CLOAD*, LOAD, GET A
REM	X	X	X	X	X	X	X	X	X	X	X	
RESET (*x,y*)								X				UNPLOT, PLOT *k,x,y*
RESTORE	X	X	X	X	X	X		X		X	X	RSTORE *n*
RESUME			X				X	X				
RETURN	X	X	X	X	X	X	X	X	X	X	X	
RSTORE *n*							X					RESTORE
SAVEA				X								STORE, PRINT=
SET (*x,y*)								X				PLOT, HPLOT, DOT, PLOT *k,x,y*
SGET											X	
SHUT											X	CLOSE
SOUND	X			X								BEEP *x,y*
SPEED			X									
STOP	X	X	X	X	X	X	X	X	X	X	X	
STORE			X									SAVEA, PRINT=
SYS		X								X		CALL, CALLM, LINK
TAB	X		X	X	X	X	X	X	X	X		COUNT *n*
TEXT			X									MODE Ø, CLEAR Ø
TOP											X	
UNPLOT									X			RESET (*x,y*), PLOT *k,x,y*
VLIN..AT			X									
VTAB (*n*)			X									
WAIT		X	X	X	X	X	X			X	X	PAUSE

Aantekeningen

Aantekeningen

Dit boek bevat een verzameling BASIC-programma's met een grote onderlinge verscheidenheid.
Het eerste deel bestaat uit educatieve programma's waarmee kinderen spelenderwijs kunnen leren lezen en rekenen. Voor de wat oudere kinderen zijn er taaloefeningen zoals ontleden en alfabetiseren opgenomen.
Bij het spelletjesgedeelte is het wel de bedoeling dat ook de kinderen aan de beurt komen en dat niet alleen vader en moeder zich over de computer ontfermen. Het zal in ieder geval een leuke ontspanning zijn na het wat serieuzere werk.
Het laatste gedeelte bevat een aantal algemene programma's die bijvoorbeeld kunnen dienen als hulp bij het berekenen van een nieuwe hypotheek of als huishoudboekje of als hulp bij de modelbouwhobby.
Met recht kunnen we spreken van programma's voor het hele gezin.

ISBN 90 201 1702 5